

AD-A034 417

PATTERN ANALYSIS AND RECOGNITION CORP ROME N Y  
THE WAVEFORM PROCESSING SYSTEM (WPS). VOLUME I.(U)  
OCT 76 P K SANYAL, D H FOLEY

F/6 9/2

F30602-72-C-0193

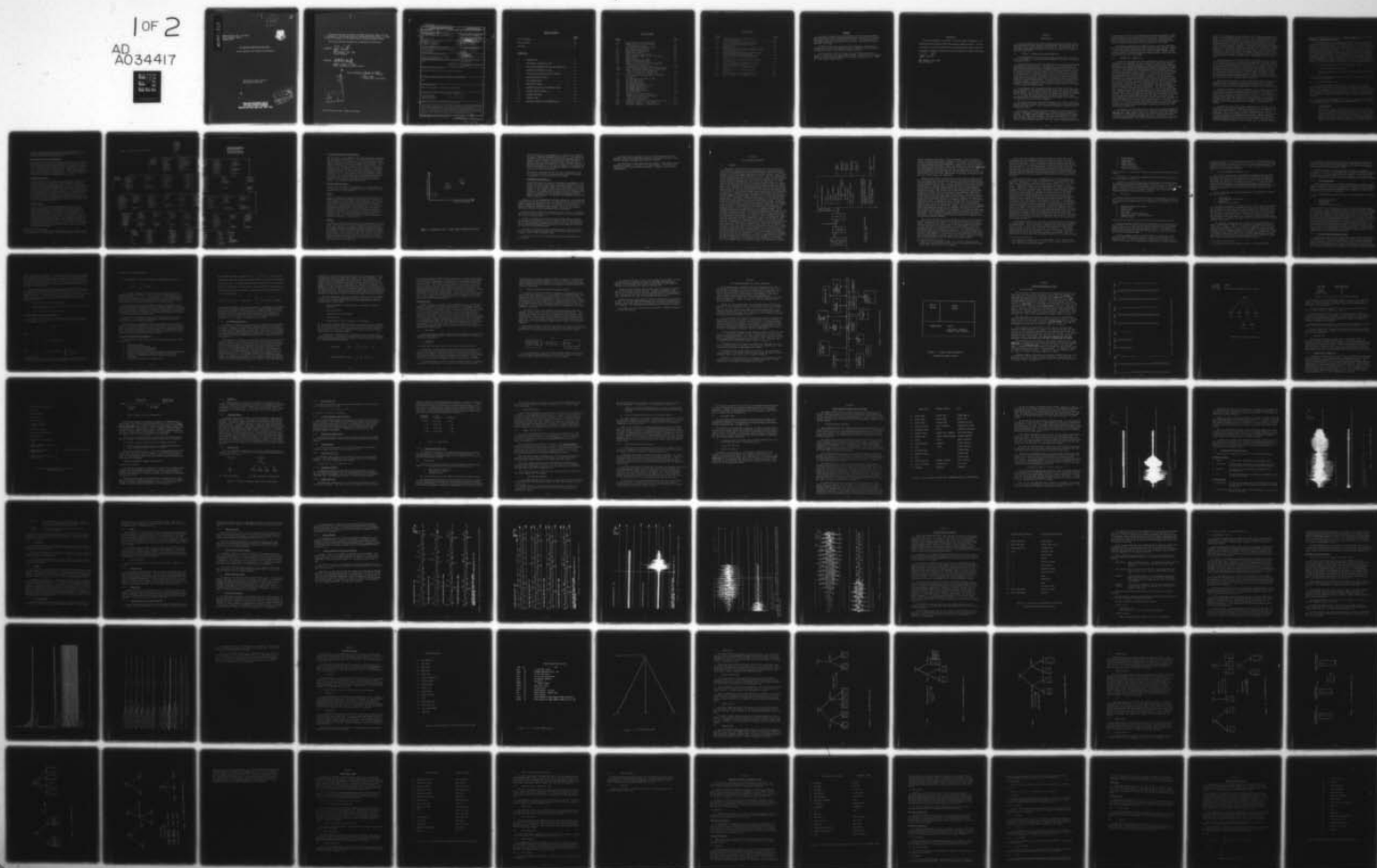
UNCLASSIFIED

PAR-76-12

RADC-TR-76-224-VOL-1

NL

1 of 2  
AD  
A034417



ADA034417

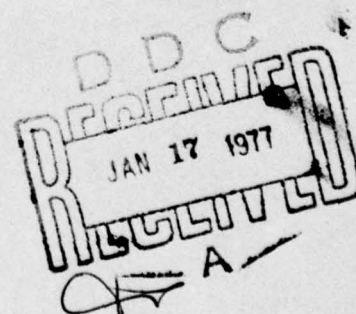
RADC-TR-76-224, Vol I (of four)  
Final Technical Report  
October 1976



THE WAVEFORM PROCESSING SYSTEM (WPS)  
Pattern Analysis and Recognition Corporation

Approved for public release;  
distribution unlimited.

ROME AIR DEVELOPMENT CENTER  
AIR FORCE SYSTEMS COMMAND  
GRIFFISS AIR FORCE BASE, NEW YORK 13441





This report has been reviewed and is approved for publication.

John Vassell

**APPROVED:**

Robert D. Knutz

FOR THE COMMANDER:

ER: *John P. Huss*  
JOHN P. HUSS

INDEXED FOR THE  
 FILE  
 FILE  
 DISSEMINATION  
 DISSEMINATION  
 BY  
 DISTRIBUTION, AVAILABILITY CODES  
 SPECIAL AGENT IN CHARGE  
 A

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER RADC-TR-76-224, Vol-1 (of four)	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) THE WAVEFORM PROCESSING SYSTEM (WPS), Volume I.	5. TYPE OF REPORT & PERIOD COVERED Final Technical Report.	6. PERFORMING ORG. REPORT NUMBER PAR 76-12	
7. AUTHOR(s) Dr. Probal K. Sanyal Dr. Donald H. Foley	8. CONTRACT OR GRANT NUMBER(s) F30602-72-C-0193	9. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62702F 55971301	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Pattern Analysis and Recognition Corporation 228 W Dominick St Rome NY 13440	10. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (ISCP) Griffiss AFB NY 13441	11. REPORT DATE October 1976	
11. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same	12. NUMBER OF PAGES 130	13. SECURITY CLASS. (of this report) UNCLASSIFIED	
14. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
15. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Same			
16. SUPPLEMENTARY NOTES RADC Project Engineer: John Vasselli, Capt, USAF			
17. KEY WORDS (Continue on reverse side if necessary and identify by block number) Waveform Analysis Feature Extraction Pattern Recognition Vol 2, A032 305 3, A029 202			
18. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report constitutes the Final Technical Report for the Waveform Processing System (WPS) developed by Pattern Analysis and Recognition Corporation for the Rome Air Development Center. The Waveform Processing System consists of (1) algorithms for performing mathematical operations on waveform data, (2) a language for extracting feature vectors from the waveform, and (3) algorithms for analyzing these vectors to develop pattern classification logic.			

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

390 101

## TABLE OF CONTENTS

	<u>Page</u>
List of Figures. . . . .	ii
List of Tables . . . . .	iii
Abstract. . . . .	iv

### Section No.

1.	INTRODUCTION. . . . .	1-1
2.	FUNCTIONAL DESCRIPTION OF WPS . . . . .	2-1
3.	WPS COMPUTER HARDWARE AND DISPLAY INTERACTION . . . . .	3-1
4.	STORAGE AND RETRIEVAL OF DATA. . . . .	4-1
5.	SINGLE WAVEFORM DISPLAY AND EDIT MODULES . . . . .	5-1
6.	MULTIPLE WAVEFORM DISPLAY. . . . .	6-1
7.	TREE MANIPULATIONS. . . . .	7-1
8.	INPUT/OUTPUT FRAME. . . . .	8-1
9.	ALGEBRAIC/CALCULUS AND ARITHMETIC FRAME . . . . .	9-1
10.	SPECTRAL ANALYSIS MODULE. . . . .	10-1
11.	SEGMENTATION FRAME. . . . .	11-1
12.	PLAYBACK FRAME. . . . .	12-1
13.	CONCLUDING REMARKS AND RECOMMENDATIONS. . . . .	13-1



## LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1-1	Options Available in the WPS System . . . . .	1-6
1-2	Scatter Plot for a 2-Class Signal Classification Problem. . . . .	1-8
2-1	Brief Functional Description of WPS . . . . .	2-2
3-1	The Hardware Layout for WPS . . . . .	3-2
3-2	Display Screen Organization . . . . .	3-3
4-1	Sample Tree Structure . . . . .	4-3
4-2	Sample ID Digit Interpretation. . . . .	4-6
4-3	Use of Temporary Symbol for the SORT Option . . . .	4-7
4-4	Sample Rates. . . . .	4-9
5-1	Waveform Displayed in Single. . . . .	5-4
5-2	Result of SELECT START TIME . . . . .	5-6
5-3	Portion of the Previous Waveform Displayed with Vertical Grid Marks. . . . .	5-11
5-4	Horizontal Grids Added to Figure 5-3. . . . .	5-12
5-5a	Cursor Positioned on Waveform for TRUNCATE FRONT. .	5-13
5-5b	Result of TRUNCATE FRONT in Figure 5-5a, with Cursor Positioned for TRUNCATE END. . . . .	5-14
5-5c	Result of TRUNCATE END in Figure 5-5b . . . . .	5-15
6-1	Three Waveforms from Same Data Set Displayed in Multi. . . . .	6-6
6-2	Ten Radar Pulses Displayed in Multi . . . . .	6-7
7-1	The LIST TREE Option. . . . .	7-3
7-2	The DRAW TREE Option. . . . .	7-4
7-3	The DELETE NODE Option. . . . .	7-6
7-4	The DELETE SUBSTRUCTURE Option. . . . .	7-7
7-5	The COMBINE NODES Option. . . . .	7-8
7-6	The APPEND NODES Option . . . . .	7-10
7-7	The APPEND FEATURES Option. . . . .	7-11
7-8	The SORT Option . . . . .	7-12
7-9	Example of the CREATE TREE Option . . . . .	7-13
10-1	Combination of Default Options in the Spectral Analysis Module. . . . .	10-6
11-1	Begin and End Parameters. . . . .	11-2
11-2	Creation of Marker Tree from Waveform Data Tree . .	11-5
13-1	Helicopter Signature: Time Domain. . . . .	13-3
13-2	Helicopter Signature: Frequency Domain . . . . .	13-4



# LIST OF TABLES

<u>Table</u>		<u>Page</u>
4-1	Common VG and Tektronix Characters and ASCII Equivalents . . . . .	4-2
4-2	Data Types and Their Associated Symbols . . . . .	4-4
4-3	Auxiliary Information Associated with Waveform Data . . . . .	4-5
4-4	Sample Rates. . . . .	4-9
5-1	Menu Options for SINGLE WAVE, EXTENDED OPTIONS, and EDIT Modules. . . . .	5-2
6-1	Menu Options for MULTIPLE WAVE INITIALIZE and MULTIPLE WAVEFORM DISPLAY Modules . . . . .	6-2
7-1	Menu of Options in the Tree Manipulations Frame . . . . .	7-2
8-1	Menu of Options in the INPUT DATA and OUTPUT DATA Frames . . . . .	8-2
9-1	Menu of Options for the ALGEBRAIC/CALCULUS and the ARITHMETIC Frames . . . . .	9-2
10-1	Menu of Options in the SPECTRAL ANALYSIS Module. . . . .	10-2
10-2	Output Options in the SPECTRAL ANALYSIS Module. . . . .	10-7
11-1	Menu of Options in the SEGMENTATION Frame . . . . .	11-6
12-1	Menu of Options in the PLAYBACK Frame . . . . .	12-2

#### ABSTRACT

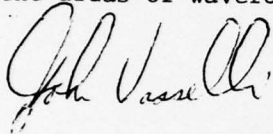
This report describes the large interactive Waveform Processing System (WPS) developed by Pattern Analysis and Recognition Corporation for Rome Air Development Center under Contract F30602-72-C-0193. The implementation of this system represents approximately 20 man-years of effort over a period of three and a half years.

The Waveform Processing System has been designed as a comprehensive research tool to aid in the complete analysis of waveforms and the design of classification logic at one dedicated computer facility.

The Final Report on WPS is in four volumes, of which this report is Volume I. Volume II is the User's Manual for WPS. Volume III is the Technical Report and User's Manual for PARLAN, the language developed for WPS. Volume IV is the Software Documentation.

#### EVALUATION

This report presents the results of the three year development of the Waveform Processing System (WPS) under Contract F30602-72-C-0193. As such, it documents the tools developed and the knowledge gained from research in the areas of waveform processing and pattern recognition feature extraction.

A handwritten signature in dark ink, appearing to read "John Vasselli". The signature is fluid and cursive, with the first name "John" and last name "Vasselli" clearly distinguishable.

JOHN VASSELLI, Capt, USAF  
Project Engineer

## SECTION 1

### INTRODUCTION

This report describes the large interactive Waveform Processing system (WPS) developed by Pattern Analysis and Recognition Corporation for Rome Air Development Center under Contract F30602-72-C-0193. The implementation of this system represents approximately 20 manyears of effort over a period of three and a half years.

#### 1.1. HISTORY OF INTERACTIVE PATTERN RECOGNITION

The implementation of the Waveform Processing System marks a milestone in the development of a complete Pattern Recognition Facility at the Rome Air Development Center.

The first major step in this development was the On-Line Pattern Analysis and Recognition System (OLPARS). The initiation of the OLPARS project was undertaken as an innovative step in bridging the gap between mathematical theory and practical application of pattern recognition technology. As with any research project, it was not known at the time whether the new approach of interactive pattern recognition would fulfill the expectations and hopes of its inventors. In those early days, as is still the case, it was recognized that the basic pattern recognition problem could best be approached by subdividing the recognition problem into the subtasks of feature extraction followed by pattern classification. It was decided that the validity of the interactive graphics approach could be initially tested by constructing an interactive system restricted to the subtasks of pattern analysis and classification; namely the OLPARS system.

It was thought that a successful OLPARS system would justify the subsequent development of interactive systems for feature definition, evaluation, and extraction. Two such feature extraction systems were initially planned; one for waveform data and the second for two-dimensional graphic data.<sup>(1),(2),</sup>

<sup>(3)</sup> As of today, the OLPARS system and its related concepts have found acceptance within the technical community and have proven their worth beyond the expectations of their initial designers.

The second phase in the long term development of a complete interactive pattern recognition facility included the implementation of the Image Feature Extraction System (IFES)<sup>(3)</sup> and its waveform counterpart, the Waveform Processing System (WPS).

In contrast to the three interactive systems, a further system was not initially recognized as a requirement for a complete pattern recognition facility. It became clearly evident during the early application of OLPARS that the basic underlying assumption, that metric proximity in the feature space reflected similarity of objects or events to be classified, was violated by certain discrete features. Thus, the Discrete Variable Analysis and



Classification System <sup>(4)</sup> was designed and is presently operational at RADC. The incorporation of this system into the RADC pattern recognition facility provides complete flexibility for any type of features and therefore we should not be inhibited from including the capability of defining discrete (type II) features within either the WPS system or the IFES system.

The Waveform Processing System manipulates only digitally stored waveform data and therefore expects all data in digital form. Routines external to WPS digitize analog waveform data and produce WPS formatted data. There is another system, the Feature Extraction System Software <sup>(7)</sup> (FESS), which operates directly on analog waveform and produces feature vectors that may be loaded either into WPS or the OLPARS.

## 1.2. CLASSICAL VS. INTERACTIVE

In order that the reader may fully understand this report, it is first necessary that he comprehend our basic philosophy of interactive pattern recognition. The classical approach to pattern recognition can largely be characterized by a researcher assuming a mathematical model which represents his data and then applying the appropriate mathematical algorithms to effect a solution. This approach is quite adequate for those problems where an appropriate model can be found; however, it is often the case that the model is not scientifically justified, rather it is selected on the basis of the existing theory found in the literature. There has always been a heavy dependence upon the so-called classical techniques (i.e., those which are described in the accepted text books) both in statistics and signal processing. It is often argued that such dependence upon the classical techniques is justified since these techniques have withstood the test of time, or, more simply stated - "they work". It is interesting to ask some fundamental questions concerning the classical methodologies. For example, examine the table of contents of any good book on statistics. The first few chapters will, no doubt, be devoted to such subjects as set theory, probability measures, random variables, and processes leading up to discussions of distribution functions. Next, the properties of several specific distributions are covered. At this point we might ask first, why are we interested in distribution functions and secondly, why these specific ones? The answer to the first question is obvious; we simply want to have a model which describes the statistical behavior of our data. The second question is a bit harder to answer. No doubt some of the classical distributions (i.e., Gaussian, uniform, Poisson, etc.) have some relation to physical processes; however, it cannot be denied that their long term acceptance is largely predicated upon their mathematical tractability.

Continuing in the statistics book, we find chapters covering estimation, sampling, hypothesis testing, correlation, confidence intervals, analysis of variance, etc. These chapters present the classical tools of the trade used extensively by statisticians throughout the years. Often we apply these techniques blindly without ever asking what it is we are after. It is not a gross oversimplification to say that the whole area of statistics seeks the

answer to the fundamental question; "What is the structure and behavior of my data?" It is interesting to note that all of the so-called classical techniques were devised long before the advent of the interactive graphic digital systems. Prior to such systems, it was necessary to formulate these neat mathematical models so that one might begin to progress toward a solution. It is no longer necessary to shackle our thinking by restricting our methodology to only the textbook techniques. Through the use of interactive graphics systems, it is possible to visually interact with one's data using the formidable power of the digital computer to answer, "What is the structure and behavior of my data?" Returning to the argument regarding the use of classical statistical techniques, we would ask if the techniques really "work", or do they only give us partial insight into the underlying data structure we seek?

The discussion above is not restricted to statistics alone, but applies to many other areas as well, such as pattern recognition and signal processing. For example, consider the prominent position held by spectral analysis in the area of signal processing. Here we refer to the generic class of all orthogonal basis and not simply to the trigonometric functions. Certainly, a researcher would have little justification other than mathematical niceties for selecting basis functions such as Fourier, Walsh, Chebyshev, Loguine, etc. unless the underlying physical process dictated its validity. Unfortunately in many signal pattern recognition problems, such as EEG analysis, the researcher has little understanding of the physical process and therefore little justification for classical spectral analysis.

Clearly, the contention is not that modeling and classical analysis are generally invalid, but rather that mathematical models can be validated through the use of interactive graphics and, furthermore, non-classical heuristic techniques should be constructable on-line to fit the data at hand.

One might be tempted to carry our argument to the opposite extreme where no classical techniques are supplied within the computer and complete reliance is placed upon the statistically-derived data base. This approach should also be subject to criticism since it would be a rare occurrence that a statistically sufficient sample data base be available. It is vitally important that insufficient statistics be replaced by human understanding of the physical process. Once again, it is through the use of interactive graphics that the researcher's a priori knowledge is inserted into the problem.

It should be clear from the above discussion that it is felt that the real potential for pattern recognition solutions lies in the proper use of interactive graphic systems. The authors specifically do not advocate the use of such systems for merely the on-line implementation of classical methodology. It is their feeling that the eventual potential for such systems can only be achieved by systems which provide both a complete graphical analysis capability and an efficient on-line language(s) for constructing processing algorithms which reflect the user's a priori knowledge and the results of graphical analysis. Thus, in their view, the two most important

functional capabilities of the WPS are: complete graphics and a versatile language for implementing heuristics.

The principal difficulty in designing interactive pattern recognition systems such as OLPARS, the Discrete Variable System, IFES, and WPS is the assurance of system flexibility. From past experience, it is known that such systems, by their nature, dynamically evolve through the experience gained in applying them to real-world problems. It would be folly to think that a first design would be totally complete and never need revision nor expansion. The WPS system design allows easy additions and/or deletions of application and graphic display programs. These requirements are not unique to pattern recognition systems since almost all interactive graphics systems generate similar requirements. However, flexibility of the data filing manager is far more subtle, requiring a deep understanding of the signal pattern recognition problem. The filing system must be designed with the following attributes:

- o It must be efficient in the use of available storage
- o It must allow rapid access to data
- o It must be structured to allow all anticipated operations upon the input data.

It is this last attribute which requires the greatest understanding of interactive pattern recognition.

### 1.3. SALIENT FEATURES OF WPS

The Waveform Processing System has been designed as a comprehensive research tool for allowing a skilled analyst the capability to digitize, filter, view, analyze, edit, compare, segment, transform, re-display, extract features, and/or design and test classification logic at one dedicated computer facility. Once an optimal classification strategy is found using WPS, the resulting logic can be implemented on special purpose hardware/firmware systems.

It should be emphasized that WPS is not just a collection or grab-bag of FORTRAN algorithms, but a sophisticated, integrated system. The system has the following features:

- o Data Management

Once the data has been properly formatted for input, the analyst never has to be aware of the data characteristics, e.g, the number of classes (nodes) in his problem (data tree), the number of waveforms at each node, or the length of each waveform. Any preprogrammed option which changes the waveform header information automatically updates the effective sampling rate, length, minimum, or maximum, etc.; both time and spectral data are handled by the



system; segment markers may be manually or algorithmically stored with each waveform; and the system can handle and distinguish between waveform and feature vector data.

- o Generality and Ease of Communication

WPS contains over 500 analyst-selectable options from a CRT terminal. Each option can be applied to any data tree. The data tree can contain anywhere from one "test" waveform to thousands of waveforms (or vectors) from many different classes. The analyst has only to specify the name of the current data set. Figure 1-1\* shows the main options available in the WPS system. Each option normally has communication messages associated with it. Via this communication the analyst can specify the parameters of his choice.

- o Background/Foreground

The executive for the WPS system has been designed to permit many complex operations to take place in the background, while the analyst continues to display and analyze data in the foreground. It is possible to designate transforms such as the complex cepstrum on data sets containing thousands of waveforms and to display the resulting transforms of the first waveform as they become available. This ability to read from a data tree while it is being written is quite advantageous. Of course, error protection routines for the "over-zealous" analyst are provided. If the transform or its associated parameter list are not satisfactory to the analyst, the transform may be easily aborted and the analyst may try again (with little of his precious time lost).

- o Advanced OLPARS

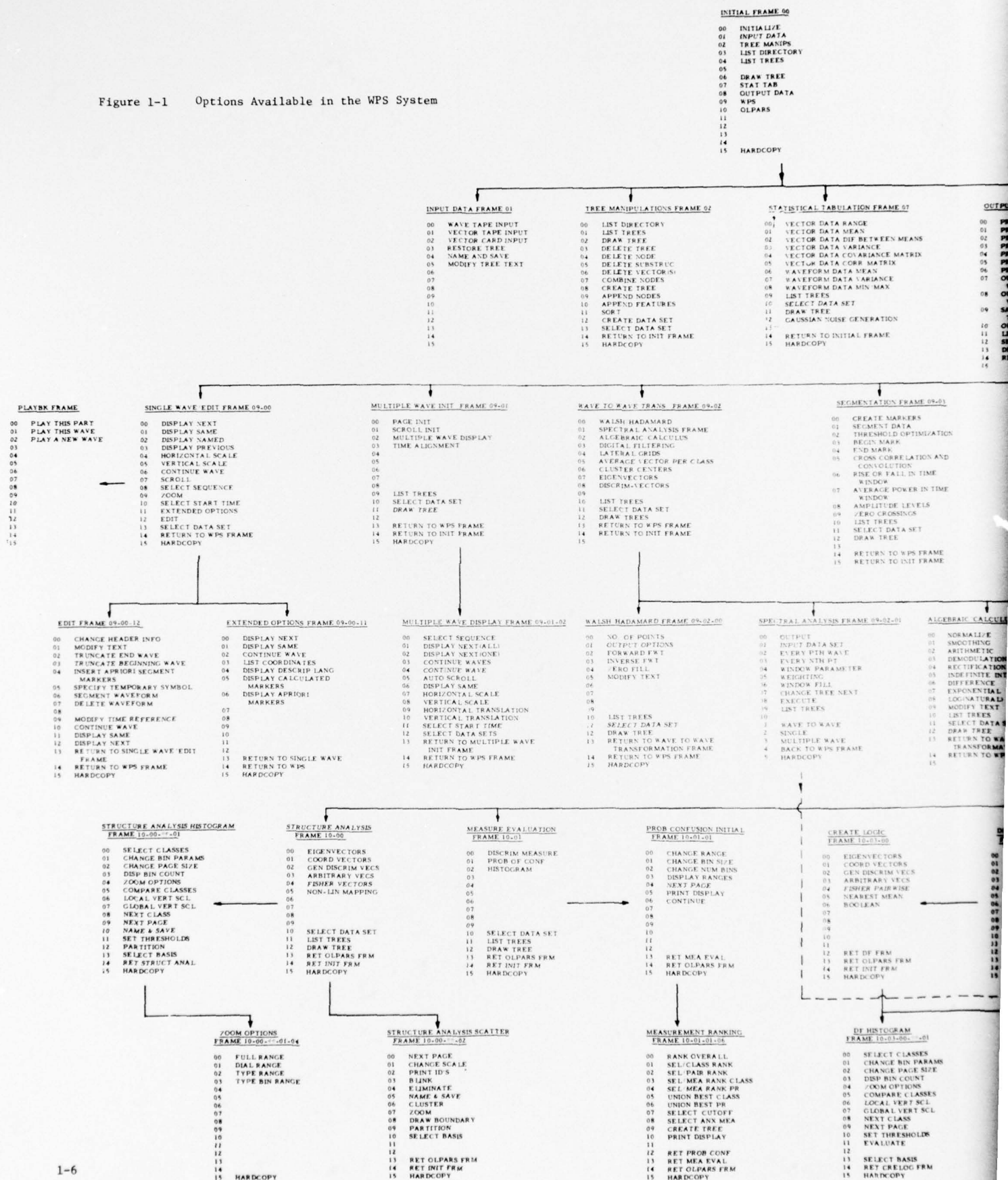
The WPS system contains the widely recognized On-Line Pattern Analysis and Recognition System (OLPARS) (8). The version of OLPARS implemented as an integral part of the WPS system is an advanced system compared to the OLPARS system described in 1969 literature by Sammon (9). For example, it is now possible to design "Boolean" or "Linguistic" type logic based on the ranges and relationships of various features. This capability is extremely important to a researcher who has become intimately familiar with the characteristics of his problem. Other additions include the partial evaluation of any portion of a logic tree, the intermixing of various logic schemes, multivariate normal Bayesian classifiers, and Mahalanobis distance.

---

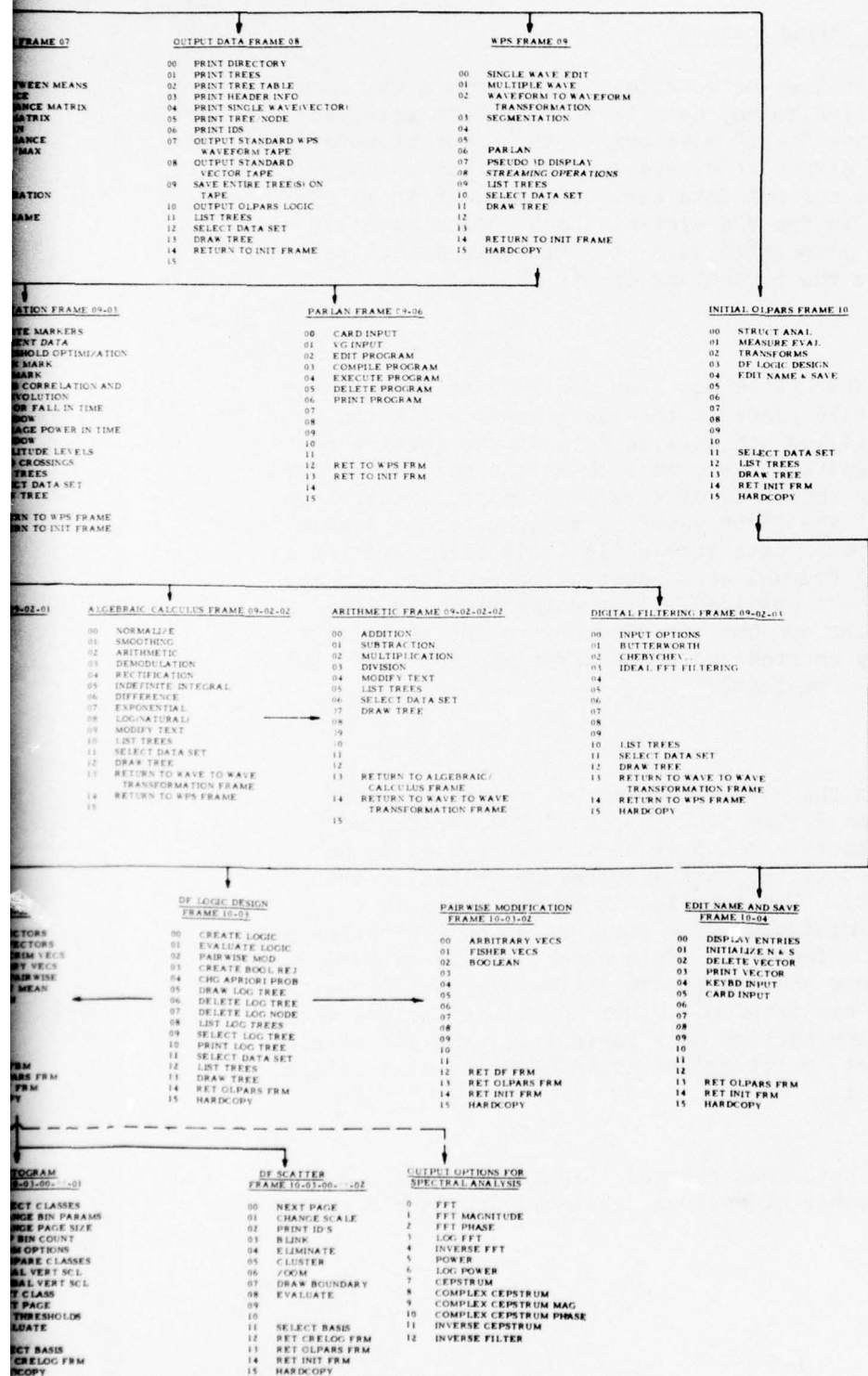
\* Although a few of the options have been changed, Figure 1-1 page 1-6 will serve to indicate the large number of options available in the WPS system.



Figure 1-1 Options Available in the WPS System



THE HARDWARE FOR WPS INCLUDES A PDP 11/45 COMPUTER, A VECTOR GENERAL DISPLAY WITH KEYBOARD AND LIGHTBUTTONS, TEKTRONIX DISPLAY AND HARDCOPY UNIT, DISK PACK, MAG-TAPE, LINE PRINTER AND SOUND PLAY-BACK SYSTEM.



- o Isomorphic/Polymorphic Transformation

The WPS system has the capability to easily handle both isomorphic and polymorphic transformations. An isomorphic transform creates one new waveform in the input data tree. All class (node) information and waveform identification numbers are preserved. Waveform header information is updated as required by the transform. An example of an isomorphic operation is the creation of the power spectrum across the length of each input waveform. A polymorphic operation generates multiple waveforms from each input waveform. An example would be the creation of power spectrums for sliding windows across the entire length of each input waveform. The on-line analyst can specify which digits in the waveform identification number are to be automatically incremented as successive "segments" are generated. Consequently, the analyst can keep abreast of the waveform-segment relationship.

- o Assembly Language Coding

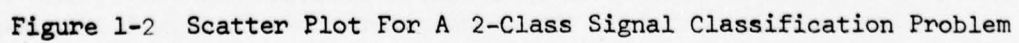
The executive, filing and data management, and memory management systems have been coded in assembly in order to minimize the time response for the on-line analyst.

- o PARLAN

Since it is virtually impossible to specify all the waveform processing algorithms which creative analysts may require, the WPS system contains its own flexible on-line language. This language, called PARLAN for Pattern Analysis and Recognition LAnguage, is unique in the sense that the user no longer has to be concerned with input and output formatting problems. Input and output are specified at execution time. For all variables in the PARLAN program beginning with "W", the analyst is asked to designate a data tree. The data tree may contain any number of waveforms, any allowable class (node) structure, and each waveform may be of different length. All this information is obtained from system tables by the program at execution time.

- o Feedback

The motivation behind incorporating signal preprocessing techniques, feature extraction routines, and classification logic design tools on the same facility is to conserve feedback and throughput time. For example, consider the scatterplot shown in Figure 1-2, page 1-8. The scatterplot is the result of a particular sequence of waveform preprocessing, feature extraction, and logic algorithms applied to a particular two-class signal classification problem. Although the analyst may observe that the class "0" is bimodal and that one sample of class "X" is imbedded as one mode of class "0", how does he proceed with an error analysis?





In the WPS system, he may designate via a light pen the "wayward" "X" and its surrounding neighbors from class "0". The analyst can immediately select an option to display this designated subset of waveforms in the multiple waveform display module (see Figure 1-1, page 1-6) and in a matter of seconds bring to bear all his experience, knowledge, and intuition in order to explain the cause of the similarity between the "X" and its neighbors. The analyst can select the original or any transformed version of his waveform to aid in his on-line error analysis.

The analyst's investigation into the cause of bimodality of the "0's" may be initiated by viewing selected subsets of waveforms corresponding to vectors from each of the modes.

o Integration and Capability

In the WPS system, all options are designed to be compatible. For example, the analyst may specify a particular segmentation process as step one in his overall signal processing problem. As a result "begin" and "end" segment markers will be inserted in the waveform header. If the analyst continues with further processing or transformation of the waveform into a feature vector, he always has the ability to designate that the remaining processing be applied between segment markers. No additional data management demands are levied on his time.

Despite all of the aforementioned features, perhaps the most significant contribution of the WPS system is its mere existence and its current use in a number of signal processing problems. The difficult hurdle of passing from a paper design merely containing a list of algorithms to a large, general purpose, integrated, dedicated, quick-response, interactive system which handles all aspects of signal processing has been surmounted.

Section 2 gives a functional description of WPS. This is a description of WPS from the user's point of view. The capabilities of WPS are covered in that section in general terms.

Section 3 discusses the specific computer configuration and analyst - CRT communication conventions. Section 4 describes the methods employed by WPS for data structures, storage, and retrieval as well as system features such as background/foreground processing and the core image file method for saving analyst-specified parameters.

Sections 5-12 discuss the single waveform display, multiple waveform display, editing, algebraic/calculus, spectral analysis, segmentation, and playback modules of WPS in specific terms.

In Section 13, we give our concluding remarks and recommendations regarding WPS.

One final remark is necessary. Due to its modular design, WPS will continue to grow as new algorithms are found via application of WPS to many practical waveform processing and recognition problems.

The Final Report on WPS consists of four volumes. This Technical Report constitutes Volume #1. Volume #2 is the User's Manual. Volume #3 is the Technical Report and User's Manual for PARLAN. Volume #4 is the Software Documentation.

## SECTION 2

### WPS FUNCTIONAL DESCRIPTION

#### 2.1. OVERVIEW

The purpose of this section is to present a functional description of the WPS system. Originally, the WPS was envisaged as a graphics-oriented system for processing waveforms and producing feature vectors that would be analyzed on the OLPARS system operating on some other machine. Later, it was decided to add all the OLPARS capabilities to WPS. Thus, the complete Waveform Processing System not only allows the user to process and view waveforms in time and frequency domains, but to analyze the waveforms in the feature space and develop classification logic, all on the same system. In this report, however, the OLPARS part has not been emphasized, because the reader can find the functional description of OLPARS in other reports, e.g. Reference 1. The description in the following pages concerns itself with the non-OLPARS part of WPS and the reader should be aware that the term WPS has been used, usually, to mean the non-OLPARS part of the Waveform Processing System. The WPS system is an enormously complex system whose purpose may be explained by Figure 2-1. Although the system was mainly intended for the purpose of defining features for classifying a waveform (i.e., a "front end" to OLPARS), one can envision the system being used for a variety of other waveform processing tasks. For example, consider a researcher who is testing new transducers for possible inclusion in an intrusion detection system. He may be interested in using WPS for the purpose of efficiently viewing the responses of the transducers under a variety of environmental conditions. His interest in WPS is simply to efficiently view and compare waveform responses. Consider the case of a second researcher who is building a waveform library representing the side lobe patterns of a set of tracking radars. He is primarily interested in viewing, editing, and labeling his waveforms in order to insure an accurate library. Still a third researcher may be interested in bandwidth compression of data to be transmitted over a communications link. His interests are characterized by a desire to "fit" the transmission signals such that fewer bits are used to adequately represent the compressed signal than were originally used to represent that signal. In this case, the researcher's interests in the WPS would involve waveform-to-waveform transformations such as orthogonal basis representations, clustering, and Karhunen-Loeve expansions. Yet another researcher might have as an end objective the design of a filter with certain observable properties in response to a selected number of input waveforms. Another important use of the WPS system will be in conducting signal detection experiments. In the case of automatic intrusion detection, the event must first be detected before the classification of the event can be accomplished. Therefore, a legitimate end objective in using WPS might be the design and testing of a signal detection algorithm. One can envision another researcher who is only interested in computing statistics along a wave (i.e., amplitude, zero crossings per time interval, etc.) or statistics across an ensemble of waves. The end objective here is



WPS

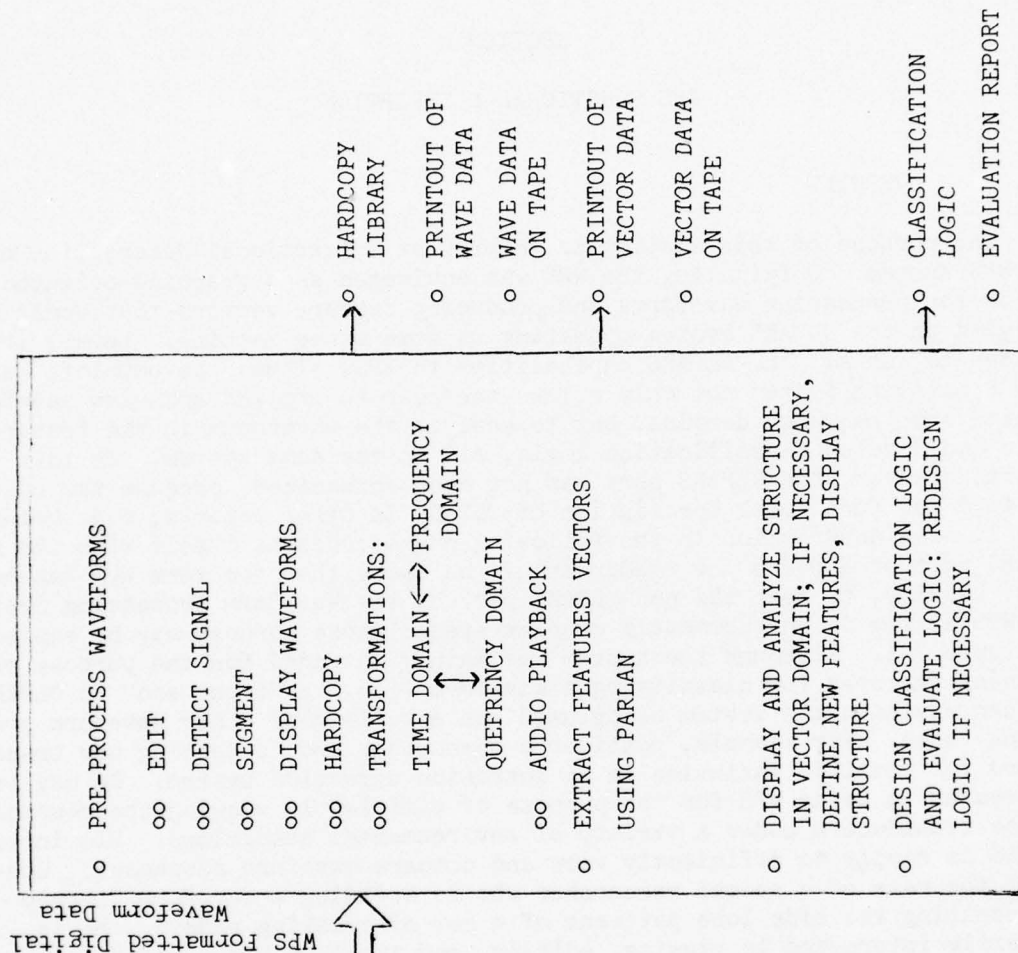


FIGURE 2-1 BRIEF FUNCTIONAL  
DESCRIPTION OF WPS

simply a catalog of statistics. A speech researcher could be attracted to use the WPS system for a variety of reasons, one of which is to design and evaluate automatic segmentation algorithms. His interest would involve using the on-line language to design algorithms to segment the continuous speech into "phonemic-like" partitions. Thus, his end objective might be segmentation. The audio playback capability of WPS will be a special help to him. Also certain transforms, e.g., Cepstrum, have been included with him in mind. This discussion could continue; however, the main thought should now be clear that the WPS can be used for a large variety of functions in addition to the main use, feature design, and evaluation.

This multi-user view of the WPS system, coupled with our basic philosophy of interactive pattern recognition systems, has laid the foundation for the system design which follows. The system has been designed from an interactive linguistic point of view. This is an important concept which deviates somewhat from the approach taken in OLPARS and the Discrete Variable System. The distinction is largely centered around the concept of preprogrammed options versus the on-line compilation of algorithms programmed during an interactive session. The nature of the pattern analysis and classification problems lent itself nicely to the concept of pre-programmed subroutines selectable by function keys and light pen operations. This approach has proven justified for both these earlier systems; however, it is the authors' opinion that the required flexibility of both the Image Feature Extraction System and the WPS system dictates the linguistic approach. It should not be interpreted that pre-programmed options, selectable via function keys and light pen, do not have their place within the WPS, since a considerable number of such options are implemented. However, such options are viewed as simply important macros in the on-line language. That is, most pre-programmed options are viewed as operators in the language and therefore they can be used as macros within the syntax of the language.\* The principal reason that such options are pre-programmed is their anticipated frequency of use by WPS researchers. This position of importance can be given to any algorithm constructed on-line by appropriately naming and saving that algorithm within the applications program library.

The result of taking this view of the WPS system has been an incredibly powerful design which will provide the flexibility required by the large variety of potential system users. It is important to note that the design does not preclude the use of the system in the pre-programmed option mode. In those cases where a researcher wishes to restrict himself to the pre-programmed options he can accomplish his goals by simply depressing keys, pointing the light pen, and entering parameters. It is not necessary to ever use the on-line language in the event that the pre-programmed options meet the user's requirements. In fact, this type of operation might be the legitimate evolutionary goal of the WPS system, which could be achieved by including all needed programs within the applications program library.

---

\* Though the on-line language, PARLAN, can be used to perform the same operation as the light button options, the actual program modules called by the light button are not used in PARLAN.

There exists still another distinction between the operational use of the system in the pre-programmed mode and the linguistic mode. The preprogrammed mode envisions the user calling for operations upon the data file to produce new data files. That is, selectable options are applied sequentially where the output from one operation produces a data file which is the input to the next operation. In the linguistic mode, the files may not actually be created;\* rather a series of operations are designed and tested on selected subsets of the data base and, when satisfied, the user may string together any previously designed algorithms or pre-programmed macros in order to define one operation which in turn can be applied to the original file outputting a final data file.

There exists one other point regarding the use of the WPS system which should be clarified at this time. A distinction must be made between "long waves" and "short waves". Although imprecise, one usually thinks of a "short wave" as being characterized by only a few number of peaks (i.e., ten or less). Pulse radar returns and electrocardiograms are examples of "short waves". Conversely, a "long wave" is characterized by a large time bandwidth product; examples of which are seismic and acoustic sensor responses and EEG waveforms. It is the authors' opinion, as reflected by the system design, that the WPS system should be impervious to this distinction. Any valid waveform operation within the WPS system can be applied to any waveform, regardless of its time bandwidth product. Obviously, there exist practical limitations on the amount of storage available, which precludes the same flexibility with "long waves" as is afforded to "short waves". The distinction between these two classes is left entirely to the user. The operations dictated by the WPS user should be selected in accord with the distribution of pertinent information within the waveforms. For some problems, the information is distributed along the waveform and is therefore considered to be "global" in nature. In contrast, for other problems the pertinent information is distributed in small regions and is therefore considered "local" in nature. When processing "global" waves, the researcher would utilize statistical computations or spectral analysis methodologies whereas for "local" waves he might be more inclined to call upon the linguistic tools within the WPS system. In either case, the system will make no distinction other than those arising from storage limitations.

The preceding describes the philosophy of design adopted for the Waveform Processing System. One is now ready to launch into the functional description of the WPS system. This will be done by examining the system from multiple points of view. To begin, the system will be envisioned as a "black box" which processes inputs to produce outputs. Thus, Section 2.2. will address the form and nature of the WPS inputs. Section 2.3. describes the pre-programmed modules of the WPS system which process the input data. This discussion will be partitioned into subsections related to the following functions:

---

\* The files may be created, but only as intermediate steps. When the complete operation is terminated, no record is kept of the intermediate files; i.e., the files do not exist anymore.



- o Manual Editing
- o Transformations
- o Segmentation
- o Signal Detection
- o Feature Extraction
- o Feature Evaluation
- o Statistical Tabulations

Next, Section 2.4. completes the "black box" discussion by describing the WPS output.

In Section 2.5. the on-line language which represents the heart of the WPS system has been described.

In Section 4, the WPS filing system is described in a functional manner with emphasis on manipulation and structure of the WPS data. A detailed software description of the filing manager is found in Volume IV of the final report. Next, the graphic subsystem is described functionally in Section 5 and 6 with emphasis on both the types of displays and the interactive relationship between the user and the WPS executive.

## 2.2. WPS INPUTS

The basic input to the WPS system is one or more waveforms, each accompanied by a list of descriptor data. Specifically, the following information will accompany each waveform and will be stored with that waveform in the WPS system:

- o Waveform Identification Number
- o Dimensionality
- o Sampling Rate
- o Time Origin
- o Descriptive Textual Annotations
- o Segmentation Marks (when appropriate)
- o Temporary Symbol

(Vector data and logic files can be inputs for the OLPARS portion of WPS).

The input data may be read in with an associated tree structure and tree name in which case the lowest order nodes are associated with the permanent symbol classes. It is felt that this concept need not be discussed in any further detail since it has been described in Section 4. The reader is probably also familiar with the data tree structure as used in both OLPARS and the Discrete Variable System.

A second category of input to the WPS system is accomplished by the researcher using the keyboard, light pen, function keys, and tablet at the display console. The use of the keyboard, light pen, and function key to input both data and control information is obvious from our OLPARS experience.

The tablet, however, offers some new possibilities.\* It can be used to input waveforms which can be traced. This operation offers a quick way of building a modest data base of digitized short waves for preliminary experiments. The tablet will also be used to control a cursor which, in turn, can be used to designate x,y coordinates on the display.

### 2.3. WPS PRE-PROGRAMMED PROCESSING MODULES

This section discusses the pre-programmed processing operations which are selectable via function keys, light pen, and keyboard input within the WPS system. The important distinction here is between these operations and those which are constructed and compiled on-line via the on-line waveform processing language which is discussed in Section 2.5. It should be noted, however, that any meaningful pre-programmed processing operation can be called as a macro operation within the on-line language and in this sense the operations discussed here can be viewed as a high level extension of our language.

The pre-programmed processing operations may be subdivided into several modules as follows:

- o Manual Editing
- o Transformation - Wave-to-Wave
- o Segmentation
- o Statistical Computations

In order to understand this subdivision, it is first necessary to comprehend the distinction between a waveform, a vector, and a string. Within the WPS, a waveform is represented as an ordered sequence of numbers which equal the amplitude of the wave sampled at specified intervals. A waveform has an initial point, a terminal point, and a specific scale which serves to relate the "distance" between sample points. In contrast, a vector is simply an ordered sequence of numbers which have no explicit "distance" measure between adjacent pairs. A string is more general than a vector since it is an ordered sequence of N-tuples.

The operations of manual editing and wave-to-wave transformations both carry a waveform into a waveform. The operation of segmentation produces a string from a waveform. That is, a waveform is segmented using initial and terminal marks which form a 2-tuple. Multiple segments along the wave are ordered from the beginning of the wave and therefore the 2-tuples are ordered producing a string. Signal Detection is a special case of segmentation since the problem of detection is equivalent to automatic segmentation of a wave, where the wave portion between the segmentation marks is considered a legitimate signal and the portion outside is considered noise.

---

\* The current version of WPS does not exploit all the possibilities.

It should be mentioned here that no distinction between waveforms and strings is made within the WPS filing system. Waveforms and strings are stored the same way and are not distinguished by the application programs whereas strings are stored as Type II data.

Before proceeding, a comment regarding the saving of a sequence of operations is in order. One recognizes the importance of storing the sequence of operations defined during an interactive session in order that the total operation can be executed off-line in a batch process. As it stands, the user is responsible for keeping track of the sequence of operations performed.

#### 2.3.1. The Editing Module

Manual editing has been intentionally separated from automatic editing since the former can have no part in any future automatic pattern recognition system. In addition, automatic editing can be conveniently included under operations such as wave-to-wave transformations, segmentation, signal detection, and on-line linguistic operations.

The following manual editing procedures are provided under the Editing Module:

- o Signal Identification
- o Time Alignment
- o Segmentation
- o Deletion of Segments, etc.

These operations will be conducted in conjunction with the Graphics Module (Section 2.7.). Signal Identification will be achieved by on-line assignment of the temporary symbol associated with the wave. This will serve to classify the signal with the generic group sharing that symbol. The filing manager will permit the sorting of files on the temporary symbol tag so that a data tree can be constructed following manual editing with the lowest order nodes corresponding to the temporary symbols assigned. It is felt that no additional signal identification procedures are needed since every signal has a unique ID which was associated with the original wave at load time. Manual time alignment can be accomplished by graphic translations using the light pen for wave point reference identification and the dials for positioning a cursor which, in turn, designates the desired translation. Similarly, manual segmentation will be accomplished by placing marks along the domain of the displayed wave via the dial-controlled cursor.

#### 2.3.2. Wave-To-Wave Transformation Module

A distinction has been made between transformations which map waveforms to waveforms from those which map waveforms to vectors since conceptually they are distinct. However, it must be noted that for certain transformations the distinction is only in the mind (and intention) of the user. For example, the coefficients resulting from a Fourier analysis may be thought of as a waveform with frequency as its domain, or they may be viewed as a feature



vector useful for classification. The distinction arises out of the intended use of the resultant coefficients. In the former case, the transformation is used for analysis in the frequency domain; whereas in the latter case the transformation produces a feature vector in which case no direct account is taken of the inter-element relationship. In fact, for the latter case, only a subset of coefficients might be selected as features which further serves to clarify the distinction.\*

The underlying purpose of the wave-to-wave transformation module is to provide a variety of pre-programmed algorithms which will be used to analyze the original waveform data. It is felt that by viewing displays of the transformed waves, the researcher may gain insight which will help him eventually solve his problem. Upon review of the possible applications of the WPS system, the set of pre-programmed wave-to-wave transformations has been subdivided as follows:

- o Classical Basis Function Transformations
- o Digital Filtering, as a light button option, has not been included due to time constraint
- o Calculus and Algebraic Transformations

#### Classical Basis Function Transformations

The options included under this heading can be utilized in two ways. First, a classical set of orthogonal basis functions can be used to "fit" a set of waveforms as follows:

$$S(t) = \sum_{j=1}^{\infty} C_j \phi_j(t)$$

where

$$C_j = \int_A^B S(t)W(t)\phi_j(t)dt$$

$$\text{and} \quad \int_A^B \phi_i(t)W(t)\phi_j(t)dt = \begin{cases} 1 & i=j \\ 0 & \text{otherwise} \end{cases}$$

---

\* In the current version of WPS, only the language, rather than any light-button option, can produce vectors from waveforms. See Section 2.5.

where  $W(t)$  is a weighting function.

An approximation to  $S(t)$  is obtained by truncating the series, i.e.,

$$\hat{S}(t) = \sum_{j=1}^N C_j \phi_j(t)$$

The "fitting" procedure could be an end in itself for the case of a bandwidth compression problem. In such a case, the coefficients of the approximated wave  $C_1, C_2, \dots, C_N$  could be used as a lower bandwidth representation of the original wave. The concept here is that the coefficients could be transmitted at a lower bandwidth than the original signal and an approximation to the signal reconstructed at the channel output.

For the purposes of pattern recognition, signal approximation holds a secondary position to discrimination since one wishes to know what characteristics of the signal classes serve to distinguish these classes. With this as the motivation, there is sometimes an interest in examining the coefficients (or functions of these coefficients) of an orthonormal expansion in the hope of finding discriminatory characteristics.

The specific orthogonal function which is implemented is the Fourier transformation. This expansion is implemented utilizing the efficiency principles of the Fast Fourier Transform.

The inclusion of additional orthonormal transformations were studied in conjunction with RADAR requirements. It is believed that little advantage is gained by studying the digital properties of orthonormal expansions independent of actual waveform problems. The requirement for additional basis functions is entirely problem-dependent and can easily be included if an actual waveform processing problem dictates their utility.

#### Calculus and Algebraic Transformations

The following is a partial list of Calculus/Algebraic operations available in WPS.

- o Indefinite Integral
- o First and Second Difference
- o Second and Third Order Exponential
- o Rectification (full and half wave)
- o Scaling (multiplication/division, constant, or prototype waveforms)
- o Addition or Subtraction (constant or prototype waveforms)
- o Normalize (to maximum value in each signal, total energy)
- o Smoothing
- o Demodulation (amplitude or frequency) etc.

The indefinite integral is given by  $F(t) = \int_A^t f(\tau) d\tau$ . The first and second differences are discrete approximations to the first and second derivatives of the wave. The second and third order exponential of  $f(t)$  are  $f^2(t)$  and  $f^3(t)$ , respectively. Rectification, Scaling, Addition, Subtraction, and Normalization are self explanatory. Smoothing replaces the waveform  $f(t)$ ,  $A \leq t \leq B$  by the function

$$F(t), A + \epsilon \leq t \leq B - \epsilon \text{ where } F(t) = \int_{t-\epsilon}^{t+\epsilon} f(\tau) d\tau \text{ and } \epsilon \text{ controls}$$

the degree of smoothing. Amplitude demodulation is accomplished by replacing  $f(t)$  by points located at the peaks of  $f(t)$ . Frequency demodulation can be applied when the number of zero crossings of the wave is large. Since the frequency of the wave may be defined as the number of zero crossings in a unit time interval, the frequency demodulation operation calculates the following wave:  $F(T)$  equals the number of zero crossings of  $f(t)$  in the interval  $T - \epsilon, T + \epsilon$  where  $\epsilon$  is determined by the analyst.

### 2.3.3. The Segmentation Module

The process of segmenting a waveform holds an important position within the WPS system since it is considered a fundamental operation in waveform processing. Segmenting involves placing pairs of marks (an initial mark and a terminal mark) along the domain of a waveform. When viewed in this manner, it is seen that the segmentation operation maps a waveform to a string where the string elements form an ordered set of two-tuples. (The ordering of the two-tuples is implicitly defined by the position of the initial mark along the waveform.)

The segmentation operation is used within the WPS system for three basic purposes. First, it is used as an end in itself when seeking solutions to Signal Detection problems. The objective here is to find algorithmic procedures for locating legitimate signals which are embedded along a waveform. The use of the WPS segmentation operations for Signal Detection will be covered in the following subsection. The second use for the segmentation operation is for the reorganization of the data base. Here one wishes to group together segments of the original waveforms in accordance with some criterion which, in turn, is implicitly defined by the segmentation algorithms. As an example of this type of use, consider a speech researcher who defines two segmentation algorithms; one for locating "vowel-like" segments and the second for locating "consonant-like" segments. After applying these



algorithms to his data base, the user may then wish to reorganize his files by creating a tree named "vowel" and a second tree named "consonant". This operation can easily be accomplished using the Sort option which will be covered under the description of the filing system. The third use of segmentation is as an intermediate operation in a computational procedure. For example, consider the use of a time window for computing a function on the segment of the wave contained within that window. Here, the time window is a segmentation which is used as an intermediate operation in the computation of the eventual function. This latter concept should become clear after reading the description of the on-line waveform processing language, since within the language the segmentation operator is given the status of a basic primitive.

Although segmentation can easily be defined within the on-line language, extensive use of certain segmentation operators is anticipated. Thus the following list of segmentation operations is pre-programmed and directly callable within the WPS system:

- o Cross Correlation or Convolution with a prototype or reference waveform.
- o Rise or Fall Time
- o Average Power in a Time Window
- o Amplitude Levels
- o Average Number of Zero Crossings in a Time Window.

All of the above operations require that the on-line user define a threshold. The initial mark is stored when the computed scalar first exceeds the threshold and the final mark is stored when the scalar falls below the threshold. This string can be stored as a "segmentation" waveform and used for future segmentation of the original waveform.

The above list of operations is self-explanatory and therefore needs little explanation. However, before leaving this subject a comment regarding cross-correlation and convolution is required. These operations are defined for continuous waveforms as follows:

$$\text{Convolution} \quad \phi_1(t) = \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

$$\text{Cross-Correlation} \quad \phi_2(t) = \int_{-\infty}^{\infty} f(\tau) g(t + \tau) d\tau$$

The difference between convolution and correlation is related to the orientation of the kernel (i.e.,  $g(t, \tau)$  function). It is well known that the most efficient digital computation of a convolution is accomplished through the use of the F.F.T. The F.F.T. of  $\phi_1(t)$  is equal to the product of the transforms of  $f$  and  $g$ . Thus the WPS implementation of convolution proceeds by first computing the FFT of  $f$  and  $g$ . Next, the product is computed in the frequency domain and finally the inverse FFT computed to yield  $\phi_1(t)$ . Cross-correlation is handled in a similar manner. The exact steps for cross-correlating  $f(t)$  with  $g(t)$  are as follows: First,  $g(t)$  is inverted to obtain  $h(t) = g(t-t)$ . Next, the convolution of  $f$  with  $h$  is computed as described above. The resultant  $\phi_1(t)$  is then inverted such that  $\phi_2(t) = \phi_1(-t)$  which produces the desired cross-correlation.

### Signal Detection

Signal Detection is a special case of segmentation since the problem of detection is equivalent to automatic segmentation of a wave where the wave portion between the segmentation marks is considered a legitimate signal and the portion outside is considered noise. Consider how the WPS system is used to solve a typical Signal Detection problem. When collecting his data, a signal detection researcher stores markers indicating the beginning and termination of an event. When his data is loaded into the WPS system, these marks are stored with the waveform and are viewed as the answer to the problem. The researcher now designs a segmentation algorithm via either the language or the segmentation module. He then causes that algorithm to be applied to his waveform(s) in the hope that the segmentation algorithm will place marks in proximity to the "a priori" marks. He can alternately display the a priori and the computed marks on the waveform and if necessary re-design the algorithm for better results.

### 2.4. WPS OUTPUT

The on-line user of the WPS system is given considerable flexibility for outputting computational results. Basically, the following two categories of output will be provided:

- o Data Files
- o Hardcopy

(The OLPARS part can output logic files and logic evaluation reports.)

The user may output to magnetic tape any data files kept by the system. It is thought that a researcher may invoke this option for a variety of reasons. For example, he may wish to store for later processing, with the WPS system, the results of partial processing, or the results of the reorganization of the data base accomplished using the Sort option. A feature vector file may be output on magnetic tape to be used as input to OLPARS.

The second category of output is hardcopy information which is generated using either the line printer or the plotter. The line printer is used to

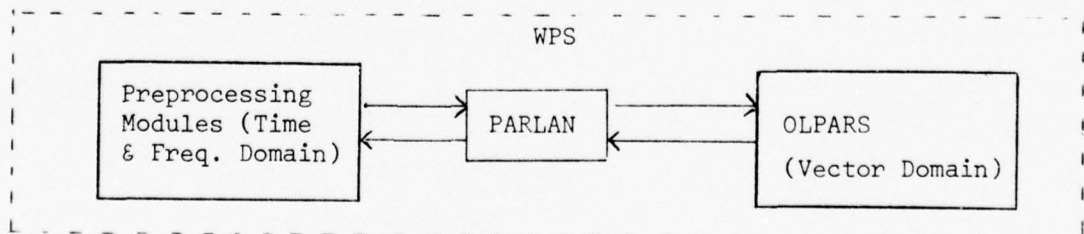
obtain hardcopy of displays (similar to OLPARS), listings of pertinent data resulting from computation, summaries of the data files (i.e., the system tables describing the data contained in the files), or the data files themselves. Waveform plots are output via the hardcopy unit associated with the Tektronix display unit.

## 2.5. THE ON-LINE WAVEFORM PROCESSING LANGUAGE (PARLAN)

No matter how large the library of segmentation, transformation, and feature extraction routines is, it is impossible to include all useful algorithms since which particular operations will be useful will be highly dependent upon the specific application at hand, i.e., on the set of waveforms and the type of processing desired. Useful operations will, in general, be discovered by the user through on-line interaction with the data. Thus, it is highly desirable that a language be supplied in which the user can designate via the keyboard general segmentation, transformation, and feature extraction operations.

The concept of the On-Line Waveform Processing Language grew quite naturally from the idea of an on-line feature definition language. Since features, in general, are extracted over segments of waves and over transformed waves, it follows that a general feature definition capability must include general waveform segmenting and waveform-to-waveform transformation capability. Thus, the attempt to design an on-line feature definition language resulted in the more general PARLAN (Pattern Analysis and Recognition LAnguage).

The status of PARLAN is that of a tie between the OLPARS portion and the non-OLPARS portion of WPS, since only PARLAN can produce feature vectors from waveform data. The following diagram describes the function of PARLAN.



As the diagram above indicates, PARLAN language programs can have both time and frequency domain waveforms and vector data as inputs and produce time and frequency domain data and vector data as outputs.



The original definition of the on-line language (called OLWPL → On-Line Waveform Processing Language) was based on very high level primitives. However, with the typical user in mind, it was decided to base PARLAN on FORTRAN, which presumably most users are familiar with.

PARLAN is fully integrated with the WPS system and therefore interacts with the WPS executive and the data base. Though the language is still referred to as being "on-line", one would normally develop the programs off-line. However, the user will tune his program on-line; i.e., make modifications and recompile as a part of his waveform analysis session.

PARLAN provides a large number of "useful" subroutines. These cannot be deleted. As the user writes his programs, with or without using these subroutines, they become additions to the subroutine library and can be used in other programs. All user-generated subroutines can be deleted by the user.

The reader will find a more detailed description of PARLAN in Volume III of the WPS Final Report.

### SECTION 3

#### WPS COMPUTER HARDWARE AND DISPLAY INTERACTION

WPS hardware includes a dedicated PDP-11/45 minicomputer with 76K of core memory, a fixed head disk with 256K words, a movable head disk with 10 million words of storage, two 9/7 track mag-tape units, a Tektronix 4002A storage tube display with a keyboard and an associated hardcopy unit, a Vector General to refresh display terminal with keyboard, a card reader, a sound playback unit, analog tape driver with associated A/D and D/A units, a line printer, and a teletype keyboard with paper tape read and punch capability. Figure 3-1, page 3-2 shows the computer facility associated with WPS.

The DEC tape and teletype units are not actively used by WPS but have been used in the development of WPS. The analog tape units and associated electronics are not directly used within WPS but are employed in an off-line mode to create WPS format digital tapes from raw analog data tapes. (See "Software Documentation for Contract #F30602-74-C-0122, Analog to Digital Conversion Routines", November 1974, submitted by Pattern Analysis And Recognition Corporation; PAR Report No. 74-22).

The Vector General (VG) display is the main CRT display for interfacing between WPS and the on-line analyst. Associated with it are a keyboard, light buttons, joystick, light pen, and data tablet. An 8K buffer in high core is reserved for the display. The VG display is continuously refreshed. However, by placing the VG buffer in high core and taking advantage of the DEC unibus partitioning architecture, minimal timing demands are imposed in the CPU for refreshing the display. The VG is a DMA device.

The VG display is organized in four regions as shown in Figure 3-2, page 3-3. The menu region contains the list of light buttons for the 16 WPS functions currently available. An option from the menu is executed by pressing the corresponding light button. Some options bring in new menus and thus one can page through the complete menu available to him. The master menu containing almost 500 options is shown in Figure 1-1, page 1-4.

The General Purpose (GP) region is available for displaying any type of visual information such as waveforms, their frequency transform, etc. This area allows the display of approximately 5500 points.

The communications region contains instructions to the user and the user's responses. Some options, when selected, request the input of various parameters and the user types them in through the keyboard.

Hardcopies of data displayed in the GP region can be obtained via the hardcopy unit, in association with the Tektronix display unit. All the analyst has to do is select the HARDCOPY menu option.

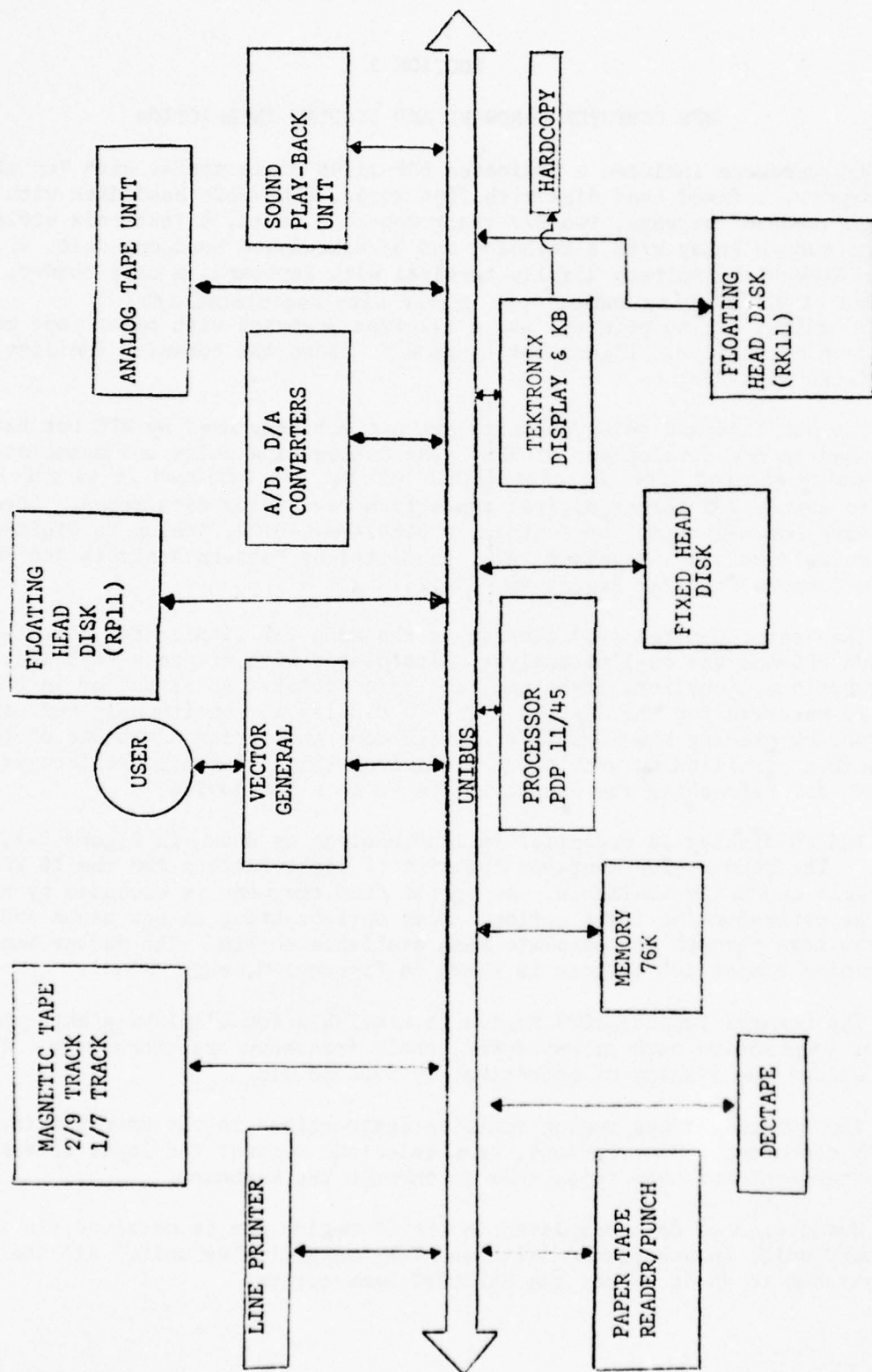


Figure 3-1 The Hardware Layout for WPS



MENU OF OPTIONS	GENERAL PURPOSE
COMMUNICATION	SYSTEM: < >  USER INPUT & EXECUTIVE MESSAGE (SELECT FUNCTION)

FIGURE 3-2 DISPLAY SCREEN ORGANIZATION  
(REGIONS NOT DRAWN TO SCALE)

## SECTION 4

### STORAGE AND RETRIEVAL OF DATA

#### 4.1. DATA STRUCTURES

WPS is capable of storing, retrieving, and manipulating many data sets. Within each data set, data can be grouped or subdivided into classes, and each class can be stored and retrieved by unique "node" names. Figure 1 illustrates a typical data set. Each data set is referred to as a "tree". Each tree name consists of six alphanumeric characters, while each node consists of five alphanumeric characters. The first character of the tree name is the "tree symbol". The remaining five alphanumeric characters of the tree name represent the name of the "senior node" of the tree. Each tree of a given data type stored in the WPS filing system must have a unique tree symbol. (Note that upper and lower case characters are considered distinct.) There are 224 different characters on the VG keyboard. Ninety-five of these characters are compatible with the Tektronix storage terminal used for hard-copy purposes. These common characters are listed in Table 4-1, page 4-3.

Node names have to be unique only within a tree, i.e., many different trees can share duplicate node names without causing any confusion. The last (sixth) character of each node is used as a plotting symbol when displaying histograms or scatter plots in the OLPARS subsection of WPS. Note that the last character of each node in Figure 4-1, page 4-3 is unique.

Blank characters are permitted for use in both tree and node names. Whenever an analyst responds to a request for a node or tree name with less than five or six characters, respectively, the characters entered are left justified and the remaining characters are filled with blanks. It is strongly recommended that analysts avoid this practice unless they are intimately familiar with WPS. For example, if a node name contains less than five characters, the plotting symbol for that node in OLPARS will be a blank.

Any node which is not divided into other nodes is referred to as a low-order node. All nodes which are neither senior nodes nor low-order nodes are referred to as intermediate nodes. All data are stored in the low-order nodes. A request for data from a senior or intermediate node retrieves data from all low-order nodes "under" the requested node. For example, a request for "EMIT3" in Figure 4-1, page 4-3 would result in the retrieval of all data from the low-order nodes named "CHANA" and "CHANB".

Although a number of different data types are permitted within WPS, only three data types are associated with tree structures. These data types (waveforms, vectors, and segment markers) along with their associated symbols are listed in Table 4-2, page 4-6 .

Table 4-1 Common VG And Tektronix Characters And ASCII Equivalents

Sr. #	ASCII	Char.	Sr. #	ASCII	Char.	Sr. #	ASCII	Char.	Sr. #	ASCII	Char.
1	40	SP	34	60	0	44	141	a	70	101	A
2	41	!	35	61	1	45	142	b	71	102	B
3	42	"	36	62	2	46	143	c	72	103	C
4	43	#	37	63	3	47	144	d	73	104	D
5	44	\$	38	64	4	48	145	e	74	105	E
6	45	%	39	65	5	49	146	f	75	106	F
7	46	&	40	66	6	50	147	g	76	107	G
8	47	'	41	67	7	51	150	h	77	110	H
9	59	(	42	70	8	52	151	i	78	111	I
10	51	)	43	71	9	53	152	j	79	112	J
11	52	*				54	153	k	80	113	K
12	53	+				55	154	l	81	114	L
13	54	,				56	155	m	82	115	M
14	55	-				57	156	n	83	116	N
15	56	.				58	157	o	84	117	O
16	57	/				59	160	p	85	120	P
17	72	:				60	161	q	86	121	Q
18	73	;				61	162	r	87	122	R
19	74	=				62	163	s	88	123	S
20	75	?				63	164	t	89	124	T
21	76	@				64	165	u	90	125	U
22	77	[				65	166	v	91	126	V
23	100	]				66	167	w	92	127	W
24	133	—				67	170	x	93	130	X
25	134	.				68	171	y	94	131	Y
26	135	l				69	172	z	95	132	Z
27	136										
28	137										
29	140										
30	173										
31	174										
32	175										
33	176										

Table 4-1 Common VG and Tektronix Characters and ASCII Equivalents



Tree Name: ERADAR  
Data Type: 63  
Tree Text: Four Emitters; Emitter #3 has 2 Channels

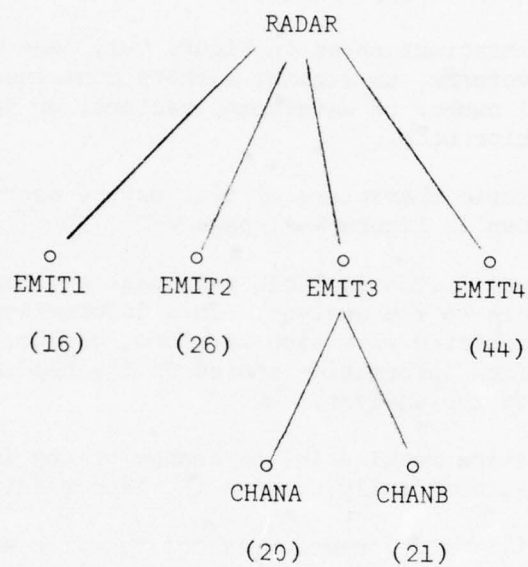


Figure 4-1 Sample Tree Structure

<u>Data Type</u>	<u>Associated Symbol</u>
Waveform	63
Vector	13
Segment Marker	90

Table 4-2 Data Types And Their Associated Symbols

The sample tree structure shown in Figure 4-1, page 4-3 also lists the number of vectors, waveforms, or segment markers contained within each low-order node. The total number of waveforms, vectors, or segment markers is listed next to the senior node.

Up to 60 alphanumeric characters of text may be associated with each tree. A sample is shown in Figure 4-1, page 4-3.

With each data type listed in Table 4-2, page 4-4, certain auxiliary information is available to the analyst. This information is either directly stored in a header associated with each waveform, vector, or segment marker or can be calculated from information stored in the header via system sub-routines transparent to the analyst.

Whenever an operation results in the change of any information described below, the operation automatically updates the header information.

Normally the auxiliary information associated with waveforms is entered with the data during input. However, this information can be modified by the analyst in the EDIT module.

#### 4.2. WAVEFORM DATA

Waveform data is stored in integer form. Each sampled value of the waveform is a single precision 16-bit integer. Consecutive samples are stored sequentially. A scale factor (a power of two) is associated with each waveform and is stored in the header. Auxiliary information associated with each waveform is available to the analyst. This information is listed in Table 4-3, page 4-5, and each item is discussed in detail below. (See "WPS Final Report Volume #IV, Software Documentation" for the exact header format).

##### 4.2.1. Identification Number (ID)

A positive ten digit number less than  $2^{32}-1$  ( $=4,294,967,295$ ) is associated with each waveform. This allows the user to uniquely identify each waveform. Various ID coding assignments have been used by different analysts. The simplest procedure assigns "1" to the first waveform collected and continues sequentially with each new waveform collected and digitized. Other analysts have coded the 10 digits into fields as shown in Figure 4-2, page 4-6. This example permits up to 42 collection sites (i.e., 0 to 41 in digits 10 and 9) and 100 vehicle types or serial numbers (i.e., 0 to 99 in digits 8 and 7). For each vehicle number and collection site there may be up to 10,000 waveforms collected (i.e., 0 to 9999 in digits 6, 5, 4, and 3).

Identification Number (ID)

Length (L)

Temporary Symbol

Waveform Text

Scale Factor (SF)

A priori Beginning  
Marker

A priori End Marker

Maximum and Minimum Values

Channel Number

Beginning Time (time domain  
data)

Beginning Frequency (spectral  
domain data)

Sample Rate (SR) (time  
domain data)

Frequency Resolution (spectral  
domain data)

} Stored with the Tree Structure

Table 4-3 Auxiliary Information Associated  
With Waveform Data



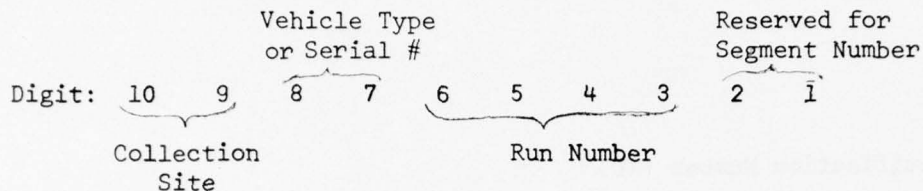


Figure 4-2 Sample ID Digit Interpretation

Note that the analyst has reserved two digits for a segment number in this example. This permits unique identification even in the case of polymorphic operations, i.e., an operation which generates multiple output waveforms from a single input waveform. Examples include automatic segmentation of a waveform into a number of sub-events (or segments) or Fourier transforming a waveform using a sliding window. As each new segment (except for the first) is generated one will be added to the first digit. In this case, 100 segments (0 to 99) can be generated for each waveform.

This ability to generate segments is more general than implied above. Whenever an analyst is performing a polymorphic operation, he is asked to:

- (1) Specify a low-order digit ( $N_1$ ) for the segment field, and
- (2) Specify a high-order digit ( $N_2$ ) for the segment field using the digit assignments shown in Figure 4-2, page 4-6.

WPS clears the field  $[N_1, N_2]$ . The first segment generated has zeros in the field. For each successive segment, a one is added to the field. Whenever the addition of a one to the digit  $N_1$  causes an overflow from the digit  $N_2$ , the polymorphic operation for the waveform under consideration is terminated, and the message

"OPERATION TERMINATED: SEGMENT FIELD EXCEEDED"

appears on the screen.

WPS permits the analyst the capability to specify his own segment field location, for each polymorphic operation, in order to permit sequences of segmentation and windowing operations. However, this generality places a burden on the analyst, i.e., he must judiciously plan, choose, and interpret the digits in the ID.

One final point: WPS does not perform time-consuming checks to determine if duplicate ID numbers exist within a tree. If duplicate ID's exist, a request to display a particular waveform ID will always result with the display of the first waveform encountered with the requested ID.

#### 4.2.2. LENGTH (L)

The number of points contained in each waveform is available to the analyst. A waveform may be as short as one point. The length of a waveform is presently limited by the data storage area on the RP02 disk which is approximately 9.6 million points. In the event that available disk storage is expanded, the software limit is 2,147,483,647 (i.e.  $2^{31}-1$ ) points in a waveform.

#### 4.2.3. Temporary Symbol

A temporary symbol consisting of a single alphanumeric character may be associated with each waveform. This symbol can be used for later sorting purposes. For example, suppose an analyst is digitizing data and that the waveforms stored on the analog tape have not been collected, recorded, and ordered by class. During A/D processing, a temporary symbol may be added to the header of each waveform. Each waveform digitized can be stored in a tree with only one low-order node as shown in Figure 4-3, page 4-7. The SORT option in the TREE MANIPULATIONS frame causes all low-order nodes to be subdivided into new low-order nodes such that all waveforms in each new low-order node have the same temporary symbol. The new node names will consist of the temporary symbol appended to the right of the four characters "NODE". (The reason for adding the characters "NODE" is to ensure that the plotting symbol used in OLPARS is not the blank character.) Figure 4-3 illustrates the results of the SORT option after it has been applied to a tree in which the low-order node "ALL" contained 55 waveforms with four different temporary symbols.

#### 4.2.4. Waveform Text

Up to 60 alphanumeric characters of text can be associated with each waveform. This text is in addition to the tree text which is associated with all waveforms in the tree.

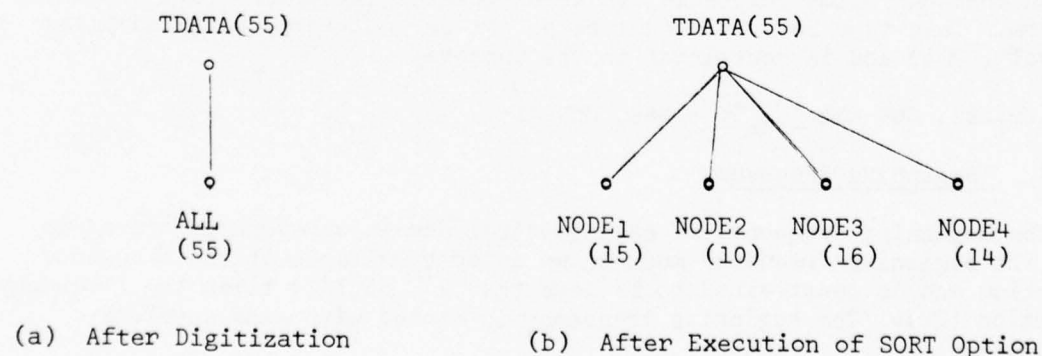


Figure 4-3 The Use Of Temporary Symbol For The SORT Option

#### 4.2.5. Scale Factor (SF)

A single scale factor is associated with each waveform. The scale factor (SF) is a power of two such that

$$SF = 2^s \text{ and } -(2^{15}) \leq s \leq 2^{15} - 1 .$$

The integer  $s$  is stored in the header.

#### 4.2.6. A priori Beginning and End Markers

One apriori beginning marker and one apriori end marker may be associated with each waveform. These markers indicate where an analyst believes that an event of interest occurs, and can be used to determine how well an automatic segmentation algorithm defines an event. The determination is made by comparing the location of the markers assigned by the algorithm with those specified apriori by the analyst. The markers can also be used to automatically segment the waveform.

#### 4.2.7. Maximum and Minimum Values

The maximum and minimum values of each waveform are stored in the header. This facilitates the process of selecting proper scaling formats for display purposes.

#### 4.2.8. Channel Number

Four bits have been reserved for the addition of channel information. Presently, channel information is not processed by WPS.

#### 4.2.9. Beginning Time ( $t_0$ )

An arbitrary time reference can be associated with each (time domain) waveform. This time reference must be an integer multiple of the sampling interval ( $\Delta t$ ) and is restricted to the interval

$$- 8,388,308 \Delta t \leq t_0 \leq 8,388,307 \Delta t .$$

#### 4.2.10. Beginning Frequency

The beginning frequency of each spectral domain waveform is stored in WPS. The beginning frequency must be an integer multiple of the frequency resolution and is constrained to be less than  $2^{16}$  (=65,536) times the frequency resolution (FR). The beginning frequency is stored with each waveform.

#### 4.2.11. Sample Rate (SR)

The sample rate (frequency), SR, is also stored with waveform data (but not with each waveform). The sample rate for all waveforms in a tree must be



identical, since the SR is associated with the tree. Presently, the sampling frequencies which can be stored are given in Table 4-4, page 4-9. If the actual sample rate used by an analyst in the digitization process is not in the table, it is suggested that a nominal value of 1000Hz be entered. Conversion factors between printed and true scaling values can be made easily by the analyst. Future modifications should consider storing the sample rate with each waveform and with floating point precision.

Beginning Frequency	End Frequency	In Steps of
.001 Hz	16.383 Hz	.001 Hz
1 Hz	16,383 Hz	1 Hz
1 KHz	16,383 KHz	1 KHz
1 MHz	16,383 MHz	1 MHz

Table 4-4 Sample Rates

#### 4.2.12. Frequency Resolution (FR)

The frequency resolution is the spacing between spectral coefficients and is analogous to the sampling interval for time domain waveforms. The frequency resolution for frequency domain data is stored with the tree structure and not with each waveform.

#### 4.3. VECTOR DATA

Vector data is stored as a sequence of single precision floating point numbers. The auxiliary information associated with vector data is:

- o Identification Number (ID)
- o Dimensionality (Length)
- o Temporary Symbol

The ID is specified in the same format as discussed for waveform data.

The dimensionality should be less than or equal to 100 points. However, certain algorithms further restrict the dimensionality because of numerical analysis considerations. For example, the pairwise Fisher linear discriminant and eigenvector algorithms constrain the dimensionality to 64 points or less.

The temporary symbol is similar to that used for the waveform data. See "WPS Final Report, Volume #IV, Software Documentation" for the exact format of the vector data.

#### 4.4. SEGMENT MARKER DATA

Segment marker or marker data store pointers to various times or frequencies of interest in a corresponding waveform data tree. Marker data is stored as sequences of double precision, 32-bit, integers. A marker tree must have the same tree structure as its corresponding waveform tree. The number of marker sequences at each node must equal the number of waveforms in each node of the corresponding waveform data tree. Normally the tree name, node names, and identification numbers associated with the marker and corresponding waveform tree are identical. Only the data type is different. Segment marker data is usually computed in the SEGMENTATION frame which is described in Section 11.

#### 4.5. CURRENT DATA SET (CDS)

As in all large software systems, the conflict of flexibility vs. ease-of-communication must be addressed. The analyst should have the capability to specify that an operation be applied to any data set. On the other side, interactive communication should be minimized when a certain data set is being used repeatedly.

To avoid this conflict, WPS uses the concept of two current datasets: one for waveforms and one for vectors. Whenever an analyst must specify a data set, he chooses the "SELECT DATA SET" option. WPS responds by requesting a data type,\* tree name, and node name. In those modules in which a data type is obvious, the first question is automatically omitted. For example, WPS assumes waveform data in the SINGLE WAVEFORM DISPLAY frame.

When the analyst types a carriage return < CR > in response to any of the questions asked in the "SELECT DATA SET" option, WPS uses the parameters stored during an earlier exercise of the "SELECT DATA SET" option.

The values stored in the CDS are changed whenever new parameters are specified in the SELECT DATA SET option. The CDS values are not changed at any other time. For example, typing in tree and node names in the DRAW TREE and INPUT WAVE options do not change the CDS.

If the analyst requests a data set which does not exist, the SELECT DATA SET option responds with the message:

"NON-EXISTENT DATASET"

In the SINGLE WAVEFORM DISPLAY frame, it is important to store information indicating not only the current data type, tree name, and node name, but also

---

\* The data type has to be indicated because though all tree names are unique under a data type, there can be two trees of the same name but of different data types.

the current waveform being displayed. This information is stored with the CDS for waveforms. This addition to the CDS permits the analyst to:

1. return to the SINGLE WAVEFORM DISPLAY at any time and immediately redisplay (via the DISPLAY NAME option) the waveform stored in the CDS; and
2. execute a transformation only on the current waveform.

The latter capability is extremely important when an analyst is searching for the correct sequence of operations. By evaluating the results of various transforms on the current waveform, the analyst can avoid time-consuming evaluations on entire data sets.

There is one exception to the SELECT DATA SET option and its subsequent effect on the CDS. This occurs in the MULTI-WAVEFORM DISPLAY frame. Since this frame permits an analyst to simultaneously display waveforms from different trees, an ambiguity as to the correct CDS arises. To avoid this possible confusion, the MULTI-WAVEFORM DISPLAY frame operates independently. That is, data set selections do not affect the values stored in the CDS. When accepted, carriage returns <CRs> in response to questions asked in the MULTI-WAVEFORM DISPLAY frame, will retrieve data parameters last used in the MULTI-WAVEFORM DISPLAY frame.

#### 4.6. BACKGROUND/FOREGROUND PROCESSING

In considering (1) the interaction between WPS and the analyst, and (2) the flow of information from one WPS module to another, a conflict involving the analyst's reasons for selecting a transformation arises.

In one case, the analyst is testing the potential usefulness of a particular transform and a particular set of associated parameters. Consequently the analyst would like to view a display of the results on one or a few waveforms as soon as possible. In the second case, the analyst has decided upon the transform and its associated parameters and wants WPS to transform a large data set.

To minimize this conflict, WPS employs the concept of foreground/ background processing. As each module, other than the SINGLE WAVEFORM DISPLAY, EDIT, and MULTI-WAVEFORM DISPLAY, are fully "debugged" via extensive researcher usage, they will be incorporated in a background mode. Once a transform is specified, the analyst can return to the SINGLE, EDIT, or MULTI-WAVEFORM DISPLAY frames which operate in the foreground and begin to view the results of the transformed data as each successive waveform becomes available.

At the present time, the INPUT WAVEFORM option has been incorporated in the background mode. Consequently the analyst can immediately begin to display waveforms as they are entered into the WPS filing system, as opposed to waiting for an entire data tape to be read.



This ability to read data from a file via one (foreground) process while the file is being written by a second (background) process is quite unique and powerful. However, some discretion on the analyst's part is necessary. "Hands-on" experience is recommended for analysts attempting to display data sets as the data sets are being written.

#### 4.7. CORE IMAGE FILES

Since any general waveform processing system must offer a large choice of parameters associated with each option, it can become quite tedious to specify these parameters. To minimize this problem WPS employs the concepts of default parameters and core image files.

If an analyst does not specify a parameter, or types a carriage return in response to a system request for a parameter, the system will use default values. Default values will be listed with the description of each option in later sections. For example, if an analyst wishes to display a single waveform and does not specify a horizontal scale, WPS will display the waveform with the following default parameters:

- o 512 intervals per line (first point of each succeeding line is the same as the last point of the preceding line),
- o 4 lines per page, and
- o Display every point.

Certainly it is impossible to select default parameters which will satisfy all applications; however, if an analyst has selected a set of parameters for a particular frame, these parameters will be saved in a "Core Image File (CIF)" when the analyst exits from the frame. Upon returning to the frame, the CIF parameters take the place of the default values. If an analyst has his own disk pack containing the WPS system, he may return days later with his CIF parameters intact.

## SECTION 5

### SINGLE WAVEFORM DISPLAY AND EDIT MODULES

The SINGLE WAVEFORM DISPLAY or SINGLE WAVE module provides an analyst with the ability to display and hardcopy any waveform in a wide variety of formats. The EDIT module provides the analyst with the capability to magnify any waveform and/or its auxiliary header information. Table 5-1 lists the 16 options associated with the function keys in the SINGLE WAVE, EXTENDED OPTIONS, and EDIT frames.

#### 5.1. DISPLAYING SINGLE WAVEFORMS

In the SINGLE WAVE frame the analyst may select any waveform data set via the SELECT DATA SET option. The selection of any one of four display options causes waveform data and its associated auxiliary information to be displayed on the Vector General (VG) refresh graphic terminal, according to the parameters associated with the HORIZONTAL SCALE, VERTICAL SCALE, START TIME, and SEQUENCE parameters.

The analyst may specify any waveform in the selected data set by typing in the waveform ID after choosing the DISPLAY NAMED option. The analyst may sequence through the data set displaying every Nth waveform by repeatedly selecting the DISPLAY NEXT option. The parameter N is set via the SELECT SEQUENCE option. Normally N is one (default value). When first attempting to display a waveform after selecting a data set, the analyst must select either the DISPLAY NAMED or DISPLAY NEXT options.

The DISPLAY PREVIOUS option permits an analyst to sequence through the waveforms in reverse order according to the value of the SEQUENCE parameter. Whenever the beginning or end of a data set is reached in the DISPLAY PREVIOUS or DISPLAY NEXT option respectively, an appropriate message is displayed in the user communication area of the VG.

The HORIZONTAL SCALE, VERTICAL SCALE, SELECT START TIME, and SELECT SEQUENCE options change the values of their corresponding parameters. However, these options do not cause the waveform to be redisplayed. This is the function of the DISPLAY SAME option. The DISPLAY SAME option is also quite useful when an analyst is returning the SINGLE WAVE frame or when he is beginning a new computer session after some earlier analysis. The execution of the DISPLAY SAME option will result in the redisplay of the previously selected waveform according to its associated display parameters stored in the Core Image File (CIF). Consequently, the tedious process of reselection of data sets and display parameters can be avoided.

The SINGLE WAVE module does not impose the restriction that a single waveform must be displayed in its entirety. If an analyst has selected his display parameters such that only a portion of the waveform is currently displayed, the CONTINUE WAVE option will display the succeeding portions of the waveform according to the selected values of the display parameters. The reader should note that the four DISPLAY options sequence from one waveform ID to another, while the CONTINUE WAVE option sequences along a single waveform.

	SINGLE WAVE	EXTENDED OPTIONS	EDIT
00	DISPLAY NEXT	DISPLAY NEXT	CHANGE WAVE ID
01	DISPLAY SAME	DISPLAY SAME	MODIFY TEXT
02	DISPLAY NAMED	CONTINUE WAVE	TRUNCATE WAVE END
03	DISPLAY PREVIOUS	LIST COORDINATES	TRUNCATE WAVE FRONT
04	HORIZONTAL SCALE	Blank	INSERT APR SEG MARKS
05	VERTICAL SCALE	DISPLAY CALC. MARKERS	SPCFY TEMP SYMBOL
06	CONTINUE WAVE	DISPLAY APRIORI MARKERS	SEGMENT WAVEFORM
07	SCROLL	Blank	DELETE WAVEFORM
08	SELECT SEQUENCE	PLAYBACK	SELECT DATA SET
09	ZOOM	Blank	MODIFY TIME REF
10	SEL START TIME	"	CONTINUE WAVE
11	EXTENDED OPTIONS	"	DISPLAY SAME
12	EDIT	"	DISPLAY NEXT
13	SELECT DATA SET	RETURN TO SINGLE	RET TO SINGLE
14	RET TO WPS FRAME	RETURN TO WPS	RET TO WPS
15	HARDCOPY	HARDCOPY	HARDCOPY

Table 5-1 Menu Options for SINGLE WAVE, EXTENDED OPTIONS, and EDIT Modules

Now let us consider the various display formats. Figure 5-1, page 5-4 is the hardcopy of a waveform displayed on the VG. The HORIZONTAL SCALE parameter was selected so that the system was instructed to "fit" the entire waveform to two lines. This START TIME was set at the beginning of the waveform. The VERTICAL SCALE was selected so that each line on the display was scaled to the maximum and minimum of the data being displayed (i.e. PAGE GLOBAL).

Certain features are automatically displayed by WPS. The line of text at the top left is the tree text. In this example, the analyst is reminded that the waveform is from a tree containing three speakers saying the word 'one.' The sampling rate is 20KHz. The speech signals were low-pass filtered at 9KHz before digitization. The line of text at the bottom left is text associated with the waveform itself. In the example, the analyst knows he is viewing a waveform associated with speaker A.

In the upper right-hand corner, the values of the selected display parameters are listed. The vertical scale (VS) is the maximum per page. The horizontal scale option FIT automatically selected the number of points per line (P/L=2276) and the number of lines per frame (L/F=2), and displayed every Pth point of the waveform (P=9). The values were selected so that the data buffer associated with the VG display would not overflow.

Note that the analyst obtains a coarse view of his data with this format: he can easily note that a DC offset existed during the A/D process and that digitization began long before speaker A uttered the word "one".

Under each line of waveform data, the integer range of the data on that line and the time associated with the beginning of each line are given. For example, line 2 begins at 1.0242 seconds and has an integer range from -3294 to 4117. As noted in Section 4, a scale factor (SF) is associated with each waveform. The exponent base 2 of the scale factor is denoted by S and is listed at the end of second line of information from the bottom of the page. Consequently the range of data on line two is

$$[-3294 \times 2^{-2}, 4117 \times 2^{-2}] = [-823.5, 1029.25]$$

The second line of text from the bottom also lists the waveform identification numbers (ID), the effective sampling rate (ESR), the number of seconds per line, the total integer range of the data being displayed, the tree and node names associated with the data displayed, and the page number. The page number is useful when the horizontal display parameters are chosen such that the entire waveform does not fit on one page.

Also note that the sampling rate as viewed by the analyst is a function not only of original sampling rate, but also of the display parameter P. The ESR for the data shown in Figure 5-1 is 20KHz/9 or 2222Hz.

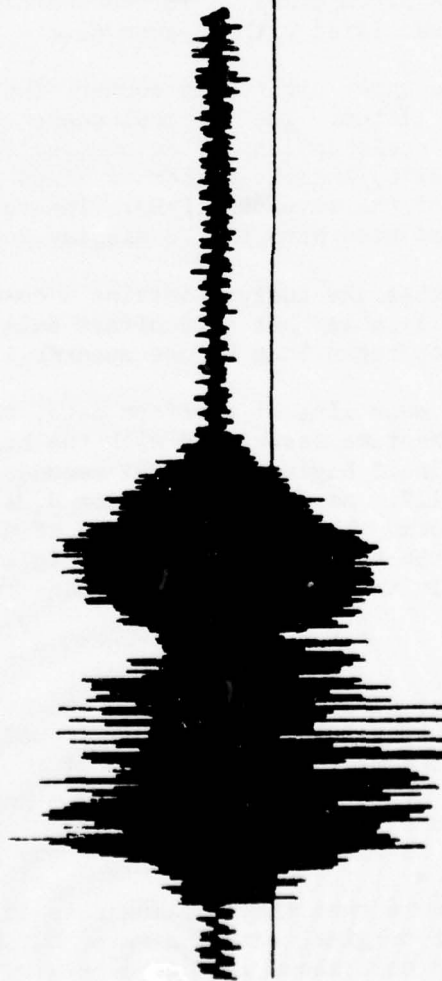


4 SPEAKERS 20KHZ/9KHZ CUTOFF "ONI."

US: M/T  
 P/I= 2276  
 L/F= 2  
 F= 9  
 FIT



LINE RANGE=352.1294 START UNITS=0s



LINE RANGE=-3294,4117 START UNITS=1.0242s  
 0000001000 ESR=2222Hz P 0242s/L R=3294,4117 TR:speech MD:perB PC:1 S=-2  
 SPEAKER A SAYING "ONE"

Figure: 5-1 Waveform Displayed in Single

Horizontal and vertical grid markers may be selected by the analyst and are discussed later. However, the zero line is automatically displayed if the waveform data being displayed range above and below zero. This provides the analyst with a quick reference.

Figure 5-2, page 5-6 shows the results of executing the SELECT START TIME and DISPLAY SAME options. The selected start time is 1.1 seconds. Note that WPS has automatically "refit" the data beginning at 1.1 seconds. All display parameters are updated as shown in Figure 5-2. An analyst can easily detect the pitch periods associated with the vowel sound in the first line of this display.

Now let us discuss the conventions used in WPS when relating the number of points (N) and the time displayed. Normally N points sampled at  $f_s$  Hertz represents  $N/f_s$  seconds of data. However, in displaying waveform data, WPS connects adjacent sampled points with straight line segments. Therefore, N points implies N-1 intervals or  $(N-1)/f_s$  seconds of data.

To avoid this confusion, WPS automatically displays N intervals per line. This requires N+1 points per line. However, the last point of one line is repeated as the first point of the succeeding line. If fewer than N+1 points of data remain, the horizontal scaling is still based on N+1 points, or equivalently,  $N/f_s$  seconds per line.

#### 5.1.1. Horizontal and Vertical Scaling

The various scale options that the user can select are as follows:

##### Horizontal Scaling

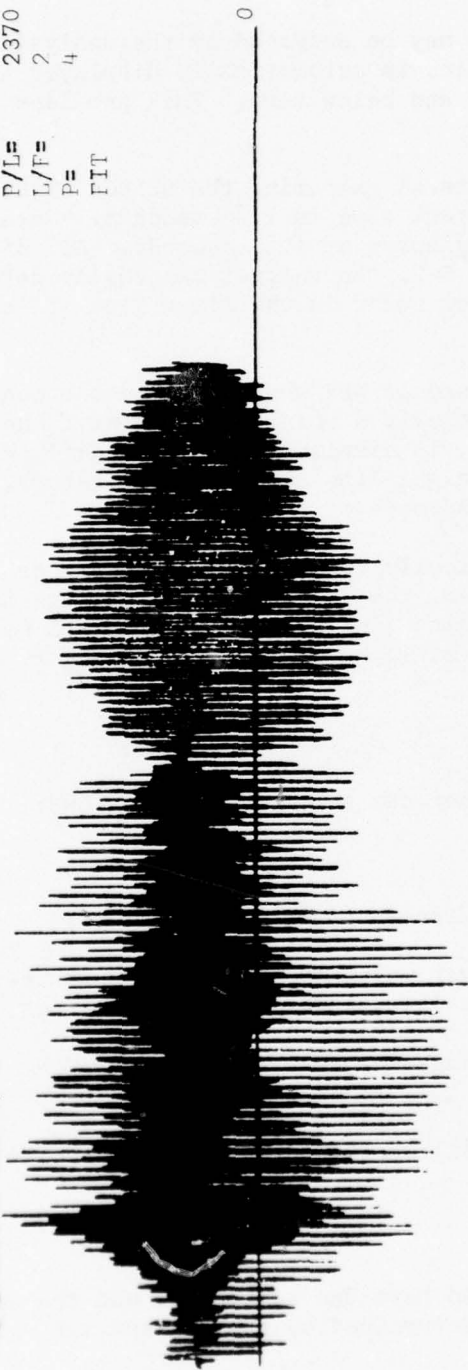
1. Default: 512 pts/line; 4 lines/frame; point sequence (P)=1.
2. Units/Line: The user enters whatever value of seconds or Hertz he wants displayed on one line, and the number of lines.
3. Explicit: The user enters N points/line, N lines/frame, and point sequence.
4. Fit: The entire waveform will be fitted on the number of lines specified by the user.

##### Vertical Scaling

1. Page Global: All the lines displayed have the same scale and the upper and lower bounds are determined by the max and min values on that page.
2. Local Scaling: The bounds for each line are determined by the max and min values on that page.

3 SPEAKERS 20KHZ/9KHZ CUTOFF "ONE"

US: M/P  
P/L= 2370  
L/F= 2  
P= 4  
FIT



LINE RANGE=-3372,4068 START UNITS=1.1s



LINE RANGE=264,1500 START UNITS=1.574s  
0000001000 ESR=5kHz 0.474s/L R=-3372,4868 TR:speech ND:perB PC:1 S=-2  
SPEAKER A SAYING "ONE"

Figure: 5-2 Result of SELECT START TIME

3. Explicit: The user enters the upper and lower bounds. Each line is scaled to these bounds. Clipping might occur if the max and min of the page exceed the bounds.
4. Waveform Global: Each line is scaled according to the max and min of the entire waveform.

Along with the specific scale options, the user has available to him a ZOOM option. With this, he can select a portion of the waveform being displayed by the light pen and that portion is displayed to the maximum scale possible on that line.

#### 5.1.2. Grid Marks

At the time the user selects scale options, he can also ask for grid marks to be displayed. The grid marks aid in estimating distances and heights (amplitudes) from the display.

The vertical grids may be specified as so many grid marks per line or as a grid mark every so many seconds or Hertz. The vertical grid marks appear as small tick marks on the base line.

The horizontal grid marks appear as lines parallel to the base line and run the full width of the display. The levels at which these lines are displayed can be specified.

#### 5.1.3. Markers

Each waveform can have stored in its "header" the positions of two (begin and end) markers. These markers are used to note an event of interest in the waveform. The markers can be inserted by a digitizing program or they may be inserted interactively in the EDIT mode of WPS. On request, these markers are displayed as vertical arrows (↑) on the waveform (DISPLAY APRIORI MARKERS).

A waveform, when recorded, can consist of a number of events separated by periods of only back-ground noise, e.g. a series of radar pulses or a series of footsteps over a line sensor, etc. Segmentation (see SEGMENTATION FRAME, Section 11) routines are available in WPS that will automatically segment the waveforms, retaining the events of interest and eliminating the rest. Before the actual segmentation is done, however, the routines create a series of markers to indicate where the waveforms are to be segmented. The analyst can have these markers displayed (DISPLAY CALCULATED MARKERS) to make sure that the segmentation criteria and parameters chosen by him will in fact retain the desired portions of the waveform.

#### 5.1.4. List Coordinates

Given the display scale and the grid marks, the analyst can guess the coordinates of any point on the waveforms being displayed. For more exact



information, he has available the LIST COORDINATES option. With this he positions a cursor on the point of interest and the coordinates of the point are displayed in the communication area.

#### 5.1.5. Scroll

The DISPLAY NEXT and CONTINUE WAVE options have been described earlier. The former brings in a similar portion of the next waveform in the file, while the latter brings in subsequent sections of the same waveform. The SCROLL option essentially simulates the repeated use of either of the above two options. The user types in a "delay", i.e. time between change of the display, so that he has sufficient time to scan the display for the features for which he is looking. The scrolling can be stopped at any time by hitting any key on the VG keyboard.

An automatic hardcopy option is also available with the SCROLL option. If this option is selected, every displayed page is hardcopied. This is an ideal method for creating hardcopy libraries. The analyst sets up the best format for the display and then selects the SCROLL option, specifying a very small "delay" time. The scrolling proceeds at a rate limited by the speed of the hardcopier, and unattended by the user.

### 5.2. EDITING WAVEFORMS

Various editing capabilities are included in the WPS. These are as follows:

#### 5.2.1. Change Wave ID

When the WPS formatted waveform data are created from analog data, an ID number is attached to each waveform. The ID numbers are sequentially generated, starting from the number one. Later, as more waveforms are generated from a single waveform, say by segmenting or by some "window operation," new ID's are generated automatically. It might happen that some ID's are non-unique. For this or other reasons, the user might choose to alter some ID's manually. This is done by typing in the old ID followed by the new ID. From then on, the particular waveform can be referred to only by the new ID.

#### 5.2.2. Modify Text

Each waveform has a text associated with it. Sometimes, it may be necessary to modify the text, for example, to include more information. Also, certain operations in WPS produce new waveforms, without associating any text with them. For posterity's sake, the user might like to add text to those waveforms. These can be done by the MODIFY TEXT option.

#### 5.2.3. Truncate Wave Front, Truncate Wave End

Very often, a small portion of a recorded waveform constitutes an event of interest, the rest being useless data. To cut down on storage and also on

subsequent processing time, it is desirable to eliminate the data that precede and follow an event of interest. The TRUNCATE options allow the user to do that.

#### 5.2.4. Segment Waveform

This option allows the user to split a waveform into two waveforms. He positions the cursor over the point of division and also supplies a new ID number. The left-hand side becomes a waveform with the original ID and the right-hand side becomes another waveform with the new ID.

Note here that this segmentation is completely directed by the user. There are other segmentation routines available (see SEGMENTATION FRAME, Section 11) that will segment waveforms automatically, based on a given criterion and the necessary parameters.

#### 5.2.5. Insert Apriori Segment Markers

The analyst can mark an event in a waveform by two markers (begin and end). To insert markers, the cursor is positioned over the desired point and a carriage return (CR) is used. The positions of the markers are stored in the header of the waveform. Other WPS routines can use the markers as an aid to processing. For example, the user may indicate that he wants the Fourier transform of the marked portion of the waveform. Or, a segmentation routine can create new waveforms out of the marked portions of waveforms.

It is to be noted that only one pair of apriori markers can be stored with a waveform, while the automatic segmentation routines can create many pairs of markers, which are stored separately.

#### 5.2.6. Specify Temporary Symbol

The purpose of the WPS is to design classification logic. To design and evaluate the logic, it is necessary to assign a class symbol to each waveform. Usually, the initial assignment of a class symbol is done at the time of preparing the WPS formatted, digitized waveforms. Either because of a mis-assignment or because no symbol was assigned, the analyst might need to specify a class symbol and this particular option allows him to do that. Any legitimate ASCII character from the VG keyboard is admissible. The class symbol is stored in the header of the waveform.

#### 5.2.7. Modify Time Reference

All time domain waveforms in the WPS format have the time coordinates of the first point of the waveform (T) stored in the header. Either because a wrong initial time is recorded or for the convenience of further analysis, the analyst may wish to change the time reference. This can be done in either of two ways. A value, positive or negative, may be typed in and that is recorded in the header. Along with this numerical value, certain letters are used to indicate the units, namely s=seconds, m=milliseconds,  $\mu$  = microseconds.

The second method of changing the time reference is for the user to position the cursor on a point on the displayed waveform where he wants to have the zero time coordinate. The system then computes the time coordinate of the beginning of the waveform, which obviously will be a negative value, and stores it in the header.

#### 5.2.8. Delete Waveform

If a waveform is found to contain no significant information, is consistently mis-classified, or for some other reason the user might find it desirable to delete certain waveforms. This is done by displaying the particular waveform and selecting the DELETE WAVEFORM option. Deletion of nodes and complete trees can be done by options under the TREE MANIPULATION frame (see Section 7).

#### 5.2.9. Further Examples Of Display And Editing

Figures 5-3 and 5-4 are examples of displays with grid markers. Both figures show a part of the waveform displayed in Figure 5-1. Note that the display parameters also have been changed by the user to points/line = 499, line/frame = 4 and P (every P<sup>th</sup> point) = 1. The vertical scale is max on page.

On Figure 5-3, the user has requested 10 vertical grid marks per line. On Figure 5-4, the user has also requested horizontal grid marks at the levels 3000, 1500 and -1500.

Figures 5-5a, 5-5b, and 5-5c show the sequence of operations in editing a waveform. The large amplitude portion of the waveform is the desired signal and the rest is supposedly background or recorder noise. In Figure 5-5a, the cursor has been positioned immediately before the signal and the portion to the left of the cursor is "truncated". Figure 5-5b shows the cursor position for a "truncate end". Figure 5-5c shows the final result. The horizontal scale has changed because the "fit" mode was in effect.

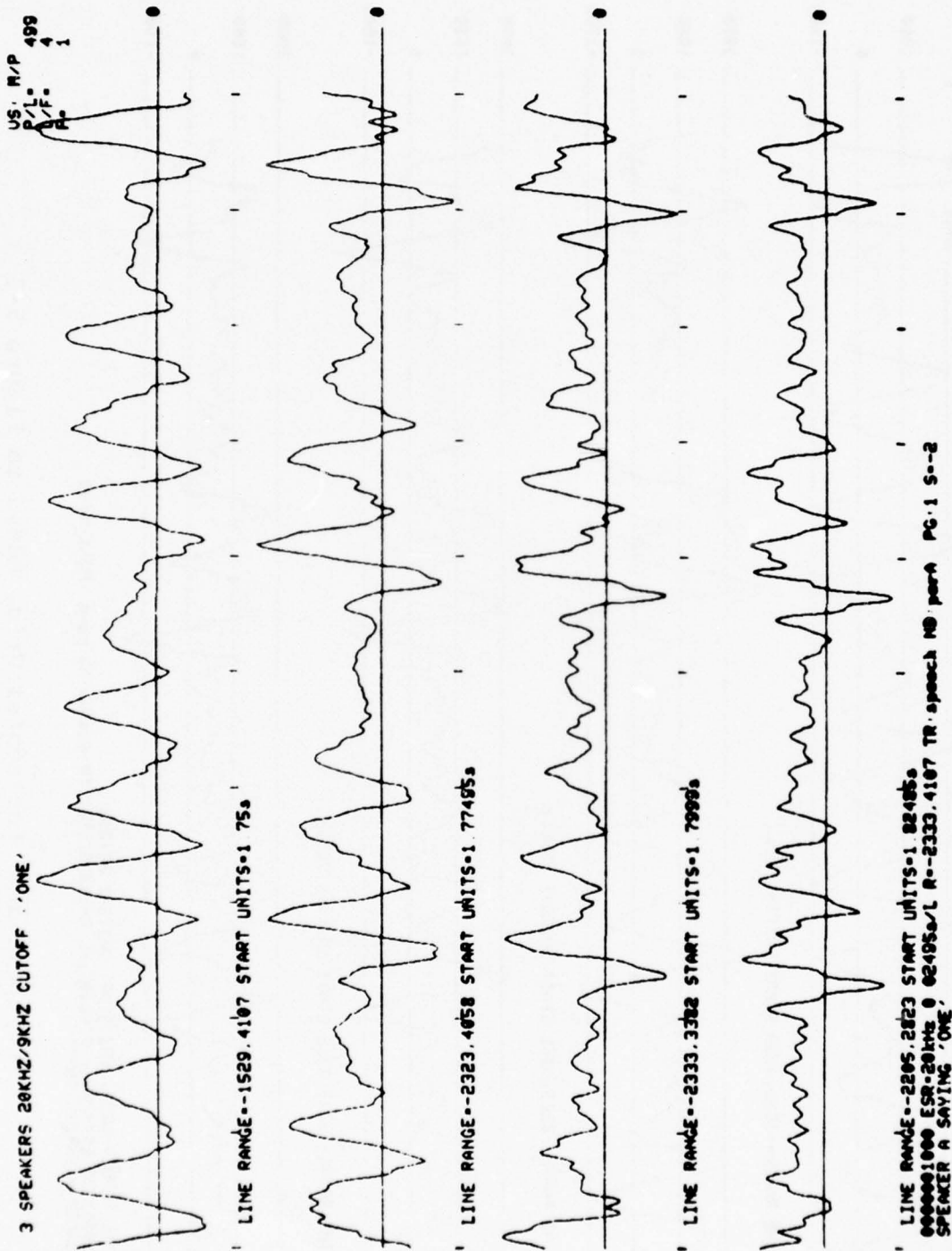


Figure: 5-3 Portion of the Previous Waveform Displayed with Vertical Grid Marks



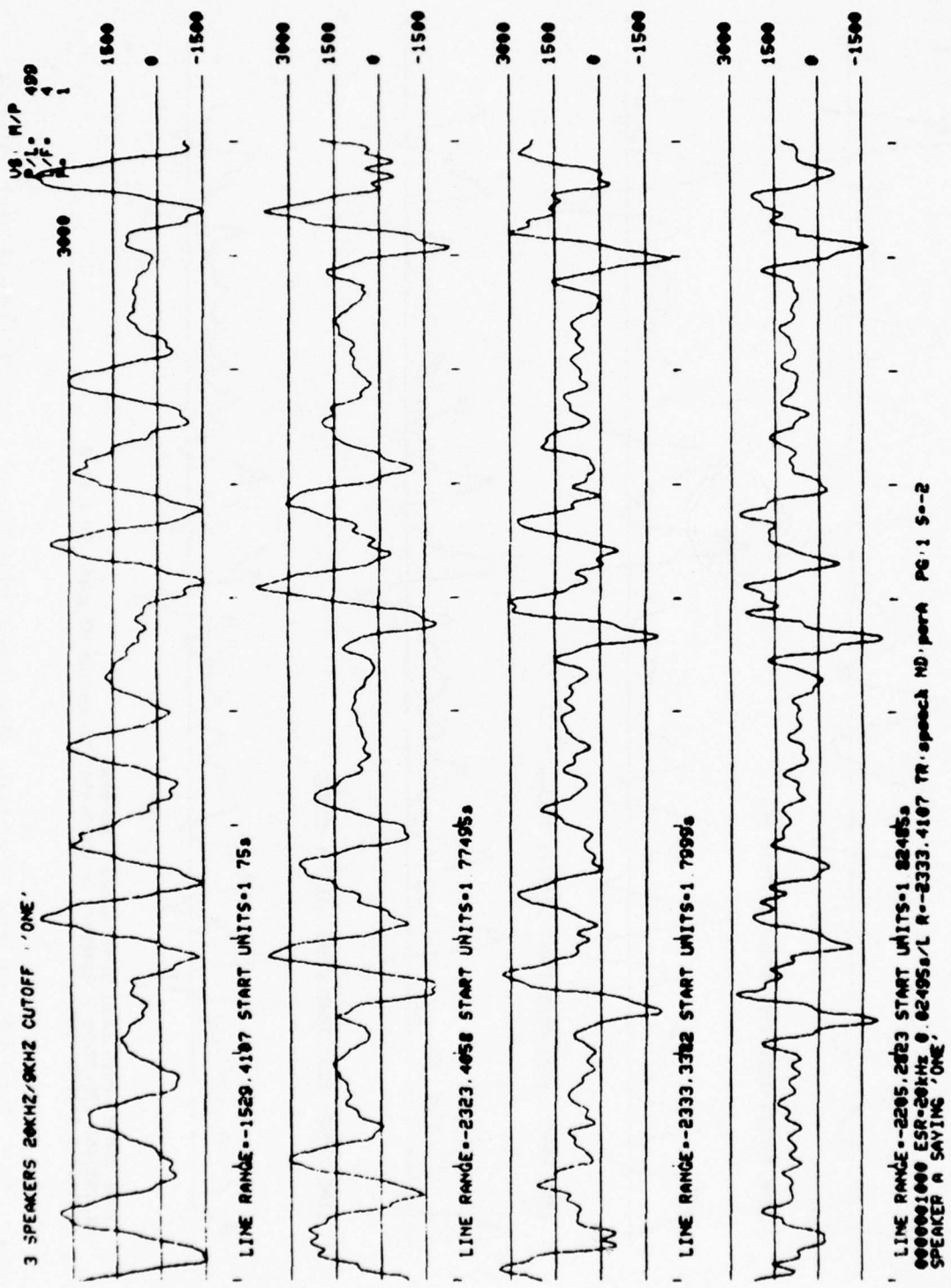


Figure: 5-4 Horizontal Grids Added to Figure 5-3

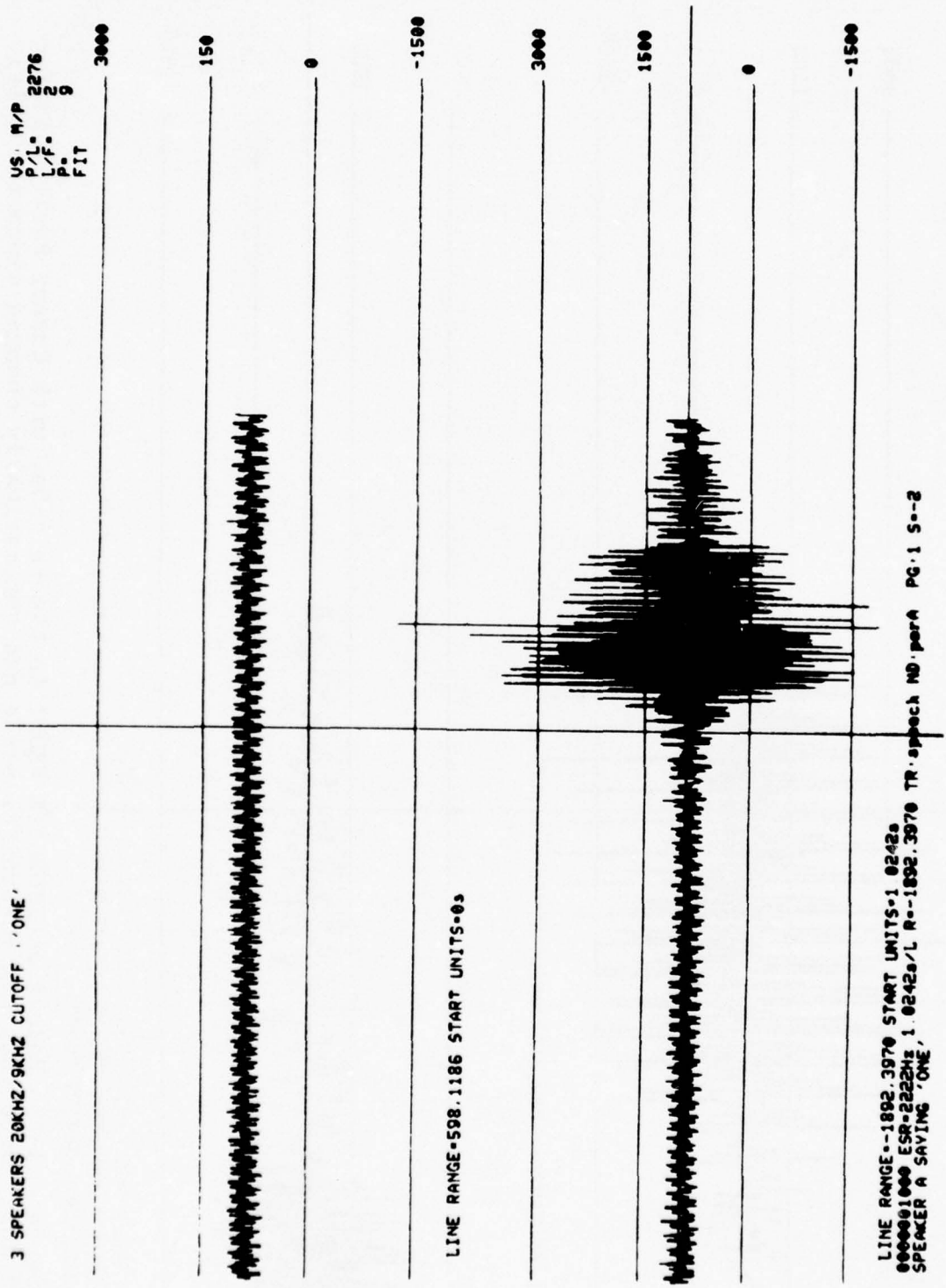


Figure: 5-5a Cursor Positioned on Waveform for TRUNCATE FRONT

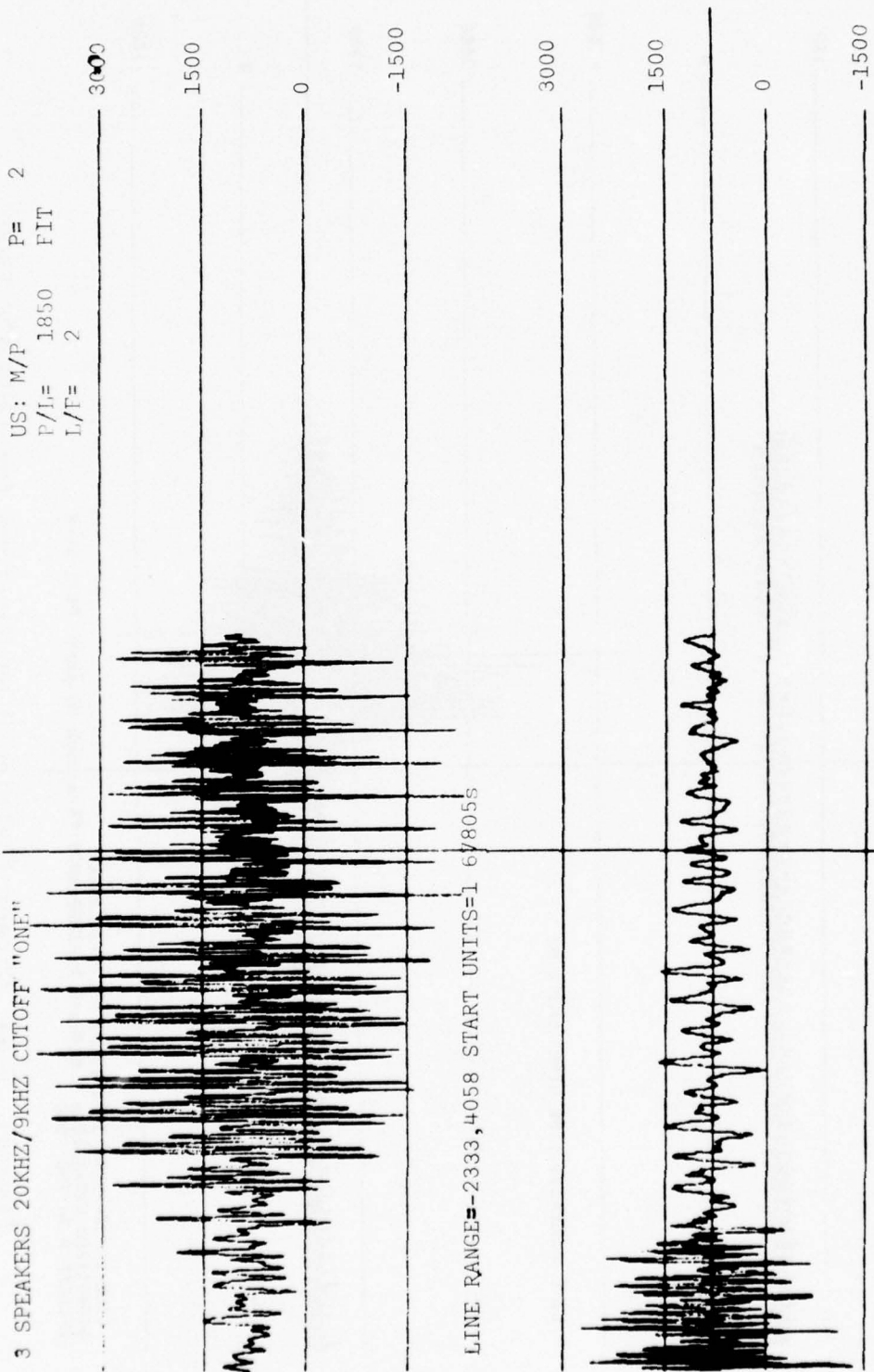


Figure: 5-5b Result of TRUNCATE FRONT in Figure 5-5a, with Cursor Positioned for TRUNCATE END. Note that the horizontal scale has automatically changed since the (fit) mode was selected.

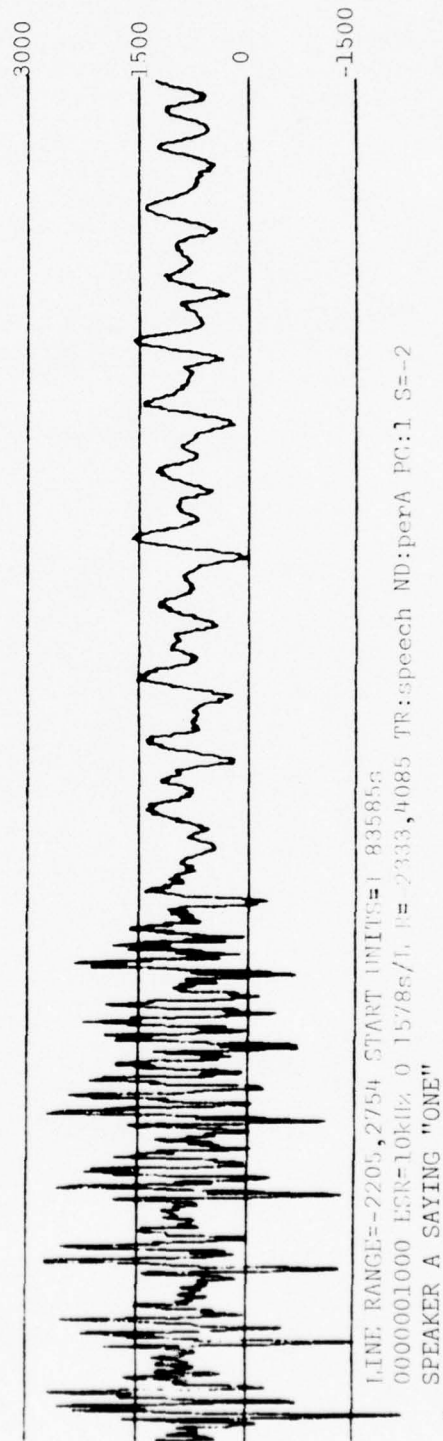
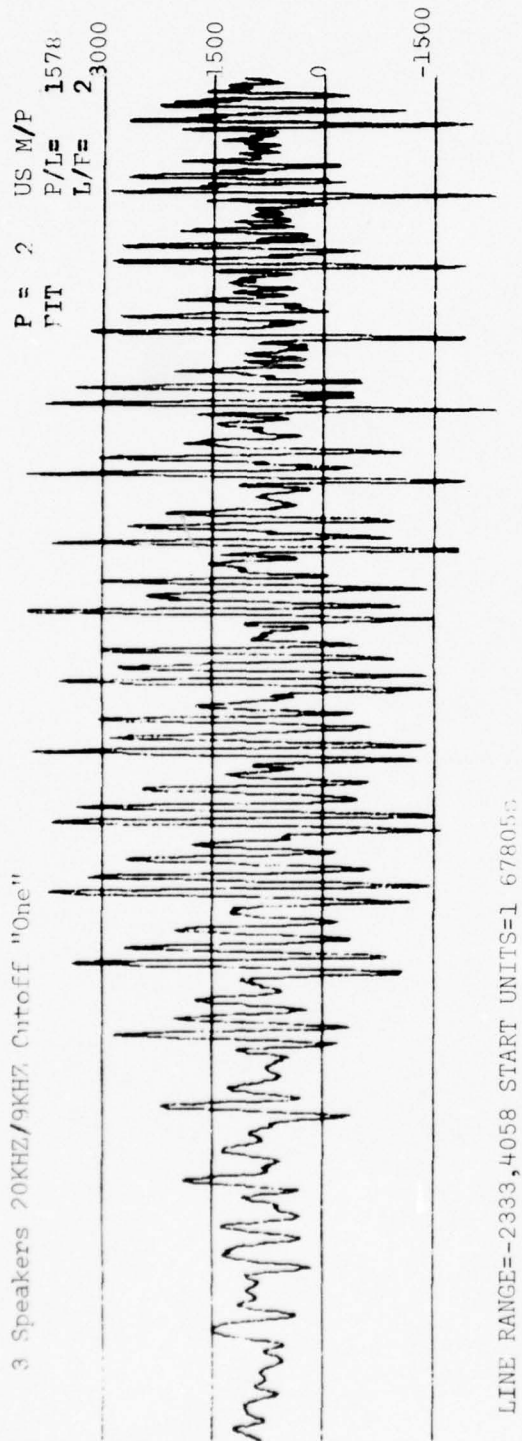


Figure: 5-5c Result of TRUNCATE END in Figure 5-5b



## SECTION 6

### MULTIPLE WAVEFORM DISPLAY

In the SINGLE WAVEFORM DISPLAY frame described in Section 5 one can view one waveform at a time. Very often, one would like to view several waveforms together for the purpose of comparison. This can be done in the MULTIPLE WAVEFORM DISPLAY module. This frame contains almost all the display options contained in the SINGLE WAVEFORM DISPLAY module, with the added feature that several waveforms can be viewed in this module. The main reason for the SINGLE WAVEFORM DISPLAY module being a separate entity is that the EDIT and PLAYBACK modules perform in conjunction with the SINGLE WAVEFORM DISPLAY module. Table 6-1, page 6-2 lists the options in the MULTIPLE WAVE INITIALIZATION frame and the MULTIPLE WAVEFORM DISPLAY frame.

The options in the MULTIPLE WAVE INITIALIZATION frame allow the user to select certain parameters for the display. These options would have logically been included in the MULTIPLE WAVEFORM DISPLAY frame, but could not be because of lack of space in that frame.

In the multiple waveform display module, up to ten different waveforms can be displayed at one time. If the analyst wants to view waveforms from one single node from a data set selected earlier in another frame, e.g. SINGLE WAVEFORM DISPLAY, then he need not employ the SELECT DATA SETS option. But to display waveforms from different nodes belonging either to the same tree or different trees, this option has to be selected. Under this option, the user proceeds to indicate the number (up to 10) of waveforms he wants displayed, and then identifies the data sets (tree names and node names) for each position of the display. If it is indicated that only one waveform is to be displayed, then the user is referred back to the SINGLE WAVEFORM DISPLAY.

After the data sets have been selected, the MULTIPLE WAVEFORM DISPLAY frame is automatically brought in, and the user selects the SEQUENCE SELECTN option to specify the start sequence number and skip sequence number for each data set selected. This option is similar to the SELECT SEQUENCE option in the SINGLE WAVEFORM DISPLAY frame. For example, suppose the user wants the waveform from TREE 1, NODE 1 to be displayed in position 5 (positions are counted from the top down of the VG screen). Suppose the file contains 100 waveforms. Let the user indicate a start sequence number of 3 and the skip sequence number of 5. Then, on successive selection of the DISPLAY NEXT option, the following waveforms will be displayed: 3, 8, 13, 18... until the end of the file is reached. The default values of the start sequence and skip sequence are 1, 1.

The various data sets involved in a multiple-wave display can be of different lengths. The DISPLAY NEXT option can be selected until all the data sets are exhausted.

Unlike the DISPLAY NEXT option, which brings in fresh waveforms from the data set, the CONTINUE WAVE brings in fresh portions of the same wave being displayed. This option may be selected as long as all the waveforms on display are not exhausted.

MULTIPLE WAVE INITIALIZE		MULTIPLE WAVEFORM DISPLAY	
00	SELECT DATA SETS		DISPLAY NEXT
01	MULTI PAGE INIT		DISPLAY NEXT (PICK)
02	MULTI SCRL INIT		CONTINUE WAVE
03	Blank		CONTINUE (PICK)
04	"		DISPLAY SAME
05	"		SCROLL
06	"		VERTICAL SCALING
07	"		HORZ SCALING
08	"		SEQUENCE SELECTN
09	"		SELECT START VAL
10	"		Blank
11	"		TRANSLATION
12	"		Blank
13	"		RET TO M-W INIT
14	RET TO WPS FRAME		RET TO WPS FRAME
15	RET TO INIT FRAME		HARDCOPY

Table 6-1 Menu Options for MULTIPLE WAVE INITIALIZE  
and MULTIPLE WAVEFORM DISPLAY Modules.

The DISPLAY NEXT (PICK) and the CONTINUE (PICK) options allow the DISPLAY NEXT and CONTINUE WAVE options to be applied to selected positions of the display; e.g. while displaying five different waveforms, the user may choose to view fresh portions of the waveform in portion 2 without changing the rest of the display. When particular positions have been picked, the DISPLAY NEXT and CONTINUE WAVE options may be used until the corresponding data sets or the waveforms are exhausted.

Unless the user specifies scale parameters for the display, default scale parameters are used. These values are 512 points/line, with every point displayed and local scaling (the max and min on the line are used to scale the vertical space allocated to each wave, the space depending on how many waveforms are displayed) in the vertical direction.

Several different scale parameters may be chosen by the two options, VERTICAL SCALING and HORIZONTAL SCALING. The vertical scale options are as follows:

1. Page Global: Scale to maximum on page. The maximum and minimum over the whole page of display are used as the upper and lower bounds.
2. Local Scaling: Scale to max on line (Default). The max and min on the line displayed are the upper and lower bounds for that line.
3. Explicit: User specified min/max. The user specifies upper and lower bounds for each line. All points of the waveform with values beyond the bounds are set equal to the appropriate bound.
4. Waveform Global: Scale to max in waveform. The upper and lower bounds for a line are the overall max and min of the waveform displayed in that line.

Different scale options may be chosen for the different lines being displayed.

The user may also choose to have horizontal grid marks displayed with any or all lines of display, at desired grid levels.

The four horizontal scale options are as follows:

1. Default values.

Points/line = 512

Point sequence p = 1, i.e. all points are displayed.

2. N units per line.

Time or frequency range to appear on one line is specified.

3. M points per line - plot every  $p^{\text{th}}$  point.
4. Fit waveform to line.

Vertical grid marks, chosen in terms of so many time or frequency units between marks or so many marks per line, can also be displayed. As with the vertical scale options, different lines of display can have different horizontal scale options.

After having selected desired scale options, the user may re-display the waveforms last displayed, by selecting the option DISPLAY SAME. Once these parameters are selected, they remain in force until they are changed by the user or until the data set is changed. Thus, all subsequent displays in the multi-wave display will incorporate these scales and other display parameters that are chosen in this frame.

By default, all waveforms are displayed from the beginning of the stored values. One can view all portions of the waveforms by repeatedly selecting the CONTINUE WAVE option. However, if one is not interested in the initial portions of some or all waveforms, he may select the SELECT START VAL option and specify a desired start unit. Different start values may be chosen for different lines. The very next time the option DISPLAY NEXT is activated, the waveforms are displayed from the indicated start values. If the CONTINUE WAVE option is selected subsequently, new portions of the waveforms are displayed, and of course, these portions will have more recent start units.

The option SCROLL simulates the effect of repeated use of the DISPLAY NEXT or the CONTINUE WAVE options. With the first option, the whole data set is scrolled; i.e. every so many seconds, new waveforms are displayed according to the specified start value and skip sequence. If the user notices any event of interest in the waveforms displayed, he can freeze the display by hitting any key on the VG keyboard.

With the second option, scrolling is done along the waveforms; i.e. every so many seconds, newer portions of the waveform are displayed. Display can be frozen by the user at any time. If the user desires, hardcopies of the displays are also automatically produced.

One of the very useful and interesting options in the multi-wave display frame is the TRANSLATION. With this option, the user can select one of the waveforms displayed and move it anywhere on the display region of the VG by turning two dials, one for the vertical translation and the other for the horizontal translation. This option allows the analyst to compare two waveforms by superimposing them.

We have said that to display waveforms from more than one data set, the user starts off with the SELECT DATA SETS option in the MULTI INIT frame. In the event that several waveforms from the same node are to be displayed, the user selects the MULTI PAGE INIT option in the MULTI INIT frame. With this option, he is able to select one node from one tree and display up to 10



waveforms from that node simultaneously. By default, consecutive waveforms starting from the first one in the node are displayed. For example, if 10 waveforms are being displayed, then the first 10 waveforms will be displayed when the DISPLAY NEXT option is selected. The second time the DISPLAY NEXT option is activated, the next 10, (i.e. 11-20), waveforms will be displayed and so on. The sequencing, start values, display scale etc. can be altered by selecting the appropriate options.

The MULTI SCROLL INIT is very similar to the MULTI PAGE INIT in that it is meant for displaying several waveforms from the same node. This particular option allows the automatic scrolling of the data set (or the waveforms), in a manner similar to that of the option SCROLL in the multi-wave display frame.

#### Examples Of Displays In Multi

Figures 6-1 and 6-2 are examples of displays in Multi.

In Figure 6-1, three waveforms from the mode "FFTMG" of the tree "1FFTMG" are displayed. The tree and node name are shown on the third line under each waveform. If there was any text associated with the tree, it would appear following the node name. The "p.1" on the third line indicates that the waveform constitutes page 1 of the waveform. If the following segments of the waveform are displayed via the use of the "CONTINUE WAVE" option, the page number is incremented appropriately. The ID number of the waveform is displayed on the second line. If there was any waveform text associated, that would be displayed on the second line, following the ID number. This particular tree had no tree text or waveform text associated with it and therefore no text appears on the display.

Also appearing on the second line are the scale factor and the vertical scale option for the waveform. For the first waveform, the scale factor is 6 ('.S=6). The vertical scale option #1 was chosen by the user ('.V=1).

The waveforms shown are, as the tree name suggests, the FFT magnitudes of three waveforms. These are, then, frequency domain waveforms. The first line of data under the waveform contains the frequency span, the amplitude span, the effective frequency resolution (EFR), and the horizontal display scale information.

The frequency span is given by the first two terms, e.g. 0 0.5Hz. Thus the frequency span is 0 to 0.5Hz.

The amplitude range is given by the next two quantities, e.g. R=0, 20699. To get the absolute values, these values have to be multiplied by the scaling factor; i.e.,  $2^S$ . For the first waveform, S=6.

The next quantity is the EFR for the waveform, e.g. EFR = 0.001953125Hz. Note that all the three waveforms have the same frequency span and EFR, but this is a coincidence.



Figure: 6-1 Three Waveforms from the Same Data Set Displayed in Multi

0.0 512s R=0.98 ESR=1000Hz	512/1		0
000000001 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.112 ESR=1000Hz	512/1		0
000000002 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.112 ESR=1000Hz	512/1		0
000000003 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.116 ESR=1000Hz	512/1		0
000000004 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.106 ESR=1000Hz	512/1		0
000000005 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.102 ESR=1000Hz	512/1		0
000000006 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.106 ESR=1000Hz	512/1		0
000000007 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.103 ESR=1000Hz	512/1		0
000000008 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.105 ESR=1000Hz			0
000000009 TR:radar ND:radar		S=1 U=1	
		p.1	
0.0 512s R=0.115 ESR=1000Hz			0
000000010 TR:radar ND:radar		S=1 U=1	
		p.1	

Note: For reproduction purposes, the third line of text under each waveform has been included on the second line. The actual display of text is similar to that of Figure 6-1.

Figure: 6-2 Ten Radar Pulses Displayed in Multi, with the Seventh one Translated and Superimposed on the Ninth one for Comparison

Following EFR data is the horizontal scale information. The quantity "256/1" signifies that there are 256 points on the line and  $p=1$  where every  $P^{\text{th}}$  point from the waveform is displayed.

Figure 6-2 displays ten radar pulses stored under the node "radar" in the tree "radar". These are time domain waveforms. Accordingly, the first line of data gives the time span (e.g. 0,0.512S) and effective sampling rate (e.g. ESR=100Hz). This figure also shows the use of the "TRANSLATE" option. The seventh waveform (ID #7) has been translated and positioned over the 9<sup>th</sup> waveform (ID #9) for comparison.



## SECTION 7

### TREE MANIPULATIONS

In Section 4, we have described the "tree" structure of the WPS data. Waveforms or vectors from a class are grouped under "low nodes". Several "low nodes" can be grouped under other intermediate nodes, which in turn can be grouped under other intermediate nodes. At the top of the tree structure is the "senior node".

When the WPS formatted data are first created, the tree consists of a senior node and one or more low nodes. The analyst, from his knowledge of the data, may re-structure the data via the options in the TREE MANIPULATIONS frame. The options in this frame are listed in Table 7-1. The descriptions of the options follow.

#### 7.1. LIST TREES

This option lists all the data trees on the VG screen. The list includes the names of the trees, the data types (i.e. waveforms, vectors or marker trees) and the text associated with the trees. The list helps the user check whether the desired data tree exists on the system and also whether there are non-essential data that should be deleted to create space for new data trees. Figure 7-1 is an example of such a list.

For convenience, this option is included in many other frames.

#### 7.2 DRAW TREE

The tree structure of a data tree can be displayed on the VG. Figure 7-2 is an example of such a display. The structure shows all the nodes appropriately connected by straight lines, the names of the nodes, the number of waveforms or vectors (the marker data are for all practical purposes like waveform data, except that they cannot be meaningfully displayed as waveforms) associated with each low node and also the senior node. The text associated with the tree is included at the top left corner of the display.

#### 7.3 DELETE TREE

In the course of a particular user's work, he might have created too many data trees that he no longer needs. Unless the trees are deleted, they clutter the storage space on the disc. The DELETE TREE option allows the user to delete any tree by typing in the name of the tree. A series of trees may be deleted by typing in their names, each followed by a CR. The user quits the option by typing in a CR as his very first response or two CR's in sequence at any other time.

When a vector data tree is deleted, the associated mean and covariance data are also deleted from the system.

## TREE MANIPULATIONS

- 00 LIST DIRECTORY
- 01 LIST TREES
- 02 DRAW TREES
- 03 DELETE TREE
- 04 DELETE NODE
- 05 DELETE SUBSTRUCTURE
- 06 DELETE VECTORS
- 07 COMBINE NODES
- 08 CREATE TREE
- 09 APPEND NODES
- 10 APPEND FEATURES
- 11 SORT
- 12 CREATE DATA SET
- 13 SELECT DATA SET
- 14 RET TO INIT FRAME
- 15 HARDCOPY

Table 7-1 Menu Of Options In The Tree Manipulations Frame

# TREES AND ASSOCIATED TEXT BLOCKS

NAME	DTP	TEXT
radar	63	EIP TEST TAPE FOR UPS
speech	63	3 SPEAKERS 20KHZ/9KHZ CUTOFF 'ONE'
0real	63	fft real coefficients
pulse	63	this is a test pulse waveform
1imag	63	fft imaginary components
2mag	63	magnitude fft
RAMP	63	xxxxxNO TEXTxxxxx
3phase	63	phase /band limited/
4logmg	63	log magnitude fft
5cpx	63	complex cepstrum real part
6kpx	63	complex cepstrum imaginary part
7con	63	continuous angle
cpxcep	63	complex cepstrum of speech segment 01 2264 seconds/51 ms
kpxcep	63	complex cepstrum of speech segment 01 2264s for 51 ms imag

Figure: 7-1 The LIST TREE Option

3 SPEAKERS 20KHZ/9KHZ CUTOFF "ONE"

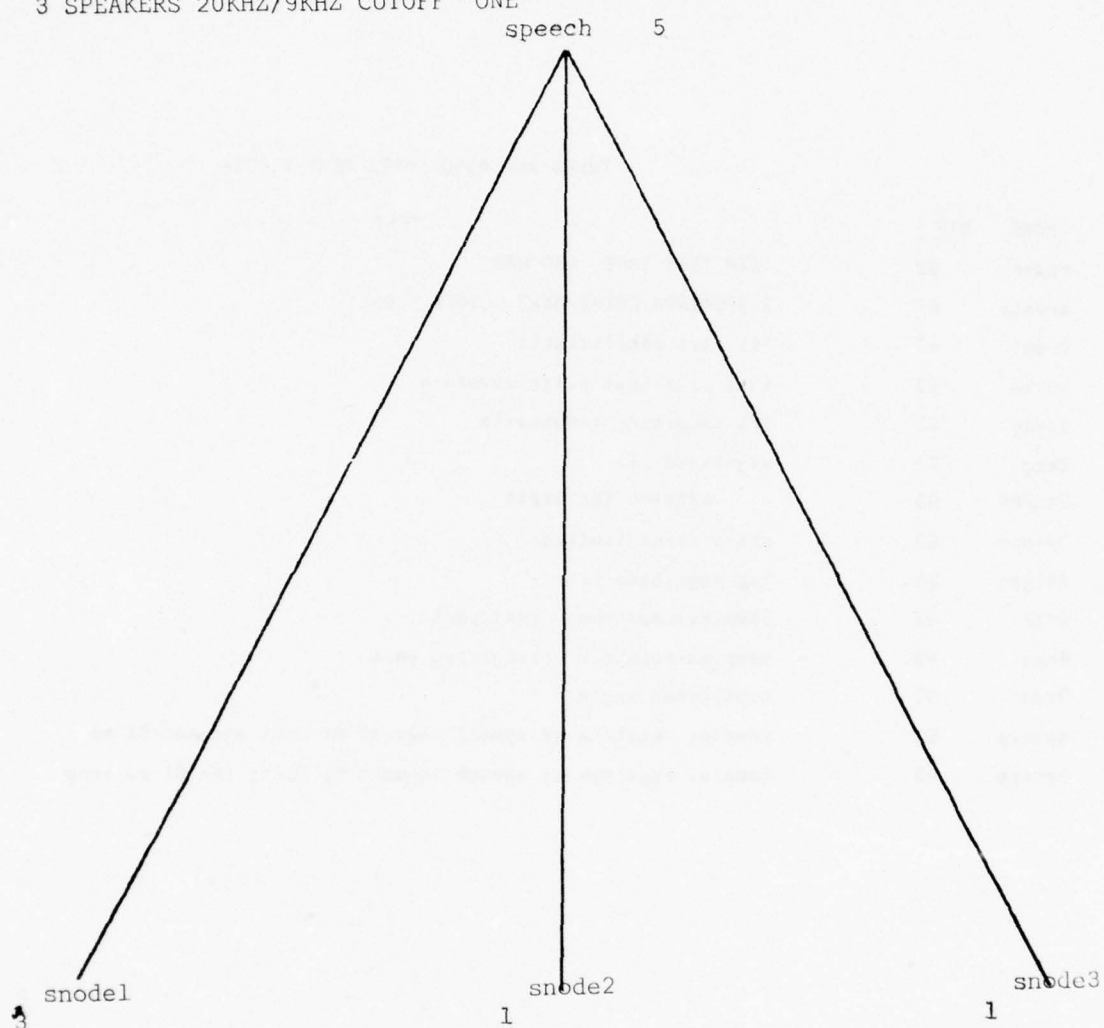


Figure: 7-2 The DRAW TREE Option



#### 7.4 DELETE NODE

This option allows the deletion of nodes from a tree. If a low node is deleted, then data from that node are wiped out from the system. If an intermediate (or the senior) node is deleted, then the data from all following low nodes are deleted. If the senior node is deleted, the effect is the same as that of deleting the tree.

When a node from a vector data tree is deleted, the vector data as well as the mean and covariance data associated with the node are deleted. Also, the mean and covariance data associated with the nodes preceding this node in the hierarchy of the tree structure are updated to reflect the loss of the vector data. Figure 7-3 explains the DELETE NODE option.

#### 7.5 DELETE SUBSTRUCTURE

This option does not delete any waveform or vector data but rather, combines the data from all the following low nodes and places them in a low node that replaces the designated intermediate node. This process eliminates the substructure that existed under the designated node.

The user supplies a name for the new low node. If the substructure under the senior node is deleted, then the resulting tree has a senior node and a single low node. The senior node is not replaced by a low node.

When the substructure from a vector data tree is eliminated, the mean and covariance data associated with the nodes in the substructure also get eliminated. However, no updating of remaining mean and covariance files is done because no vector data are lost. Figure 7-4 explains the DELETE SUBSTRUCTURE option.

#### 7.6 DELETE VECTORS

The delete option described so far allows one to "prune" the data trees in various ways. With this option, the user can delete individual vectors from the low nodes. This is similar to the DELETE WAVEFORM option in the EDIT frame.

To delete a vector, the user types in the ID number followed by a CR. A series of vectors may be deleted by typing in their ID numbers, each followed by a CR. A sequence of two CR's indicates the end of the delete option. When a vector is deleted, all pertinent mean and covariance files are appropriately updated.

#### 7.7 COMBINE NODES

The DELETE SUBSTRUCTURE option combines all low nodes under a designated node into a low node. With COMBINE NODE, the user can selectively combine low nodes that share a common, immediate senior node; the old low nodes are deleted. When the option is applied to vector data, a new mean and covariance file is created for the combined low node. Figure 7-5 explains this option.

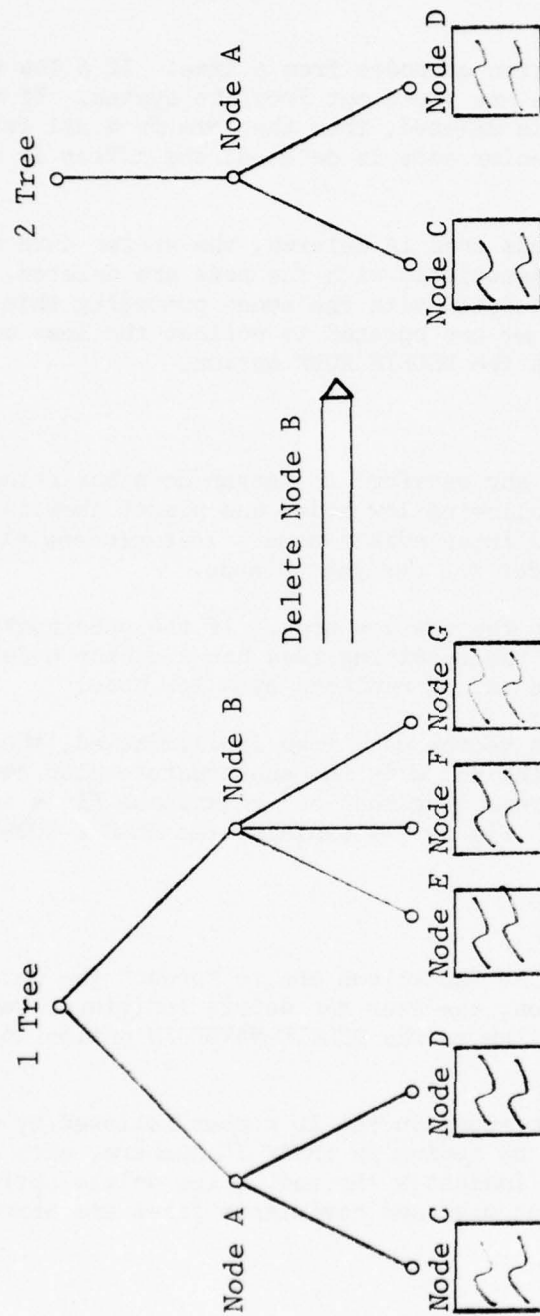


Figure: 7-3 The DELETE NODE option

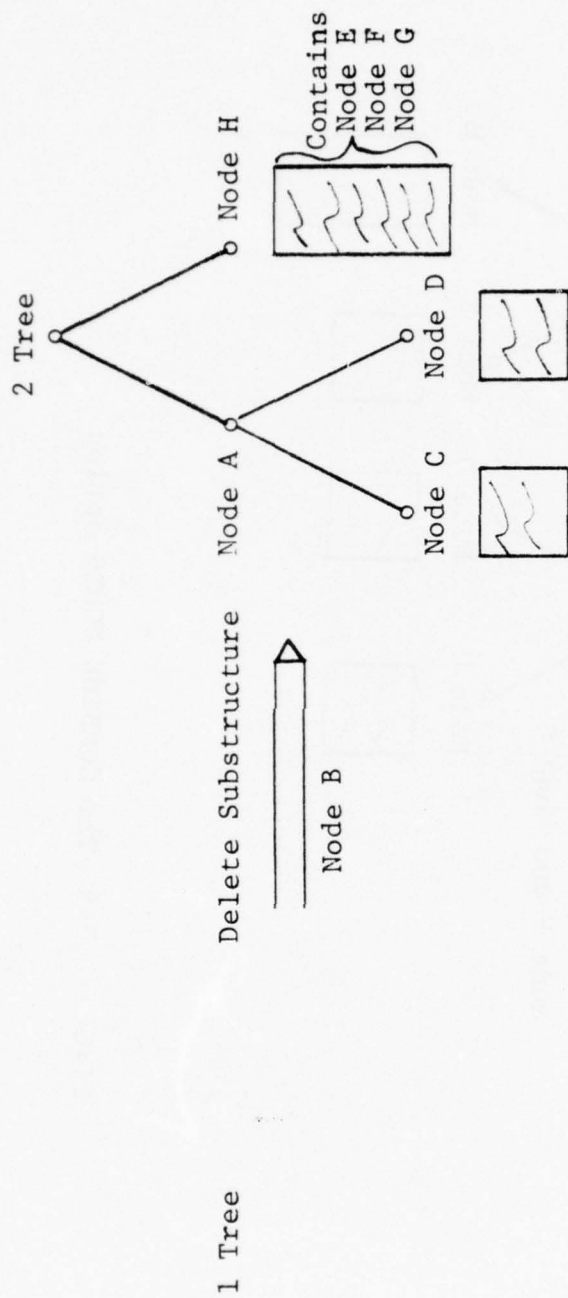


Figure: 7-4 The DELETE SUBSTRUCTURE option

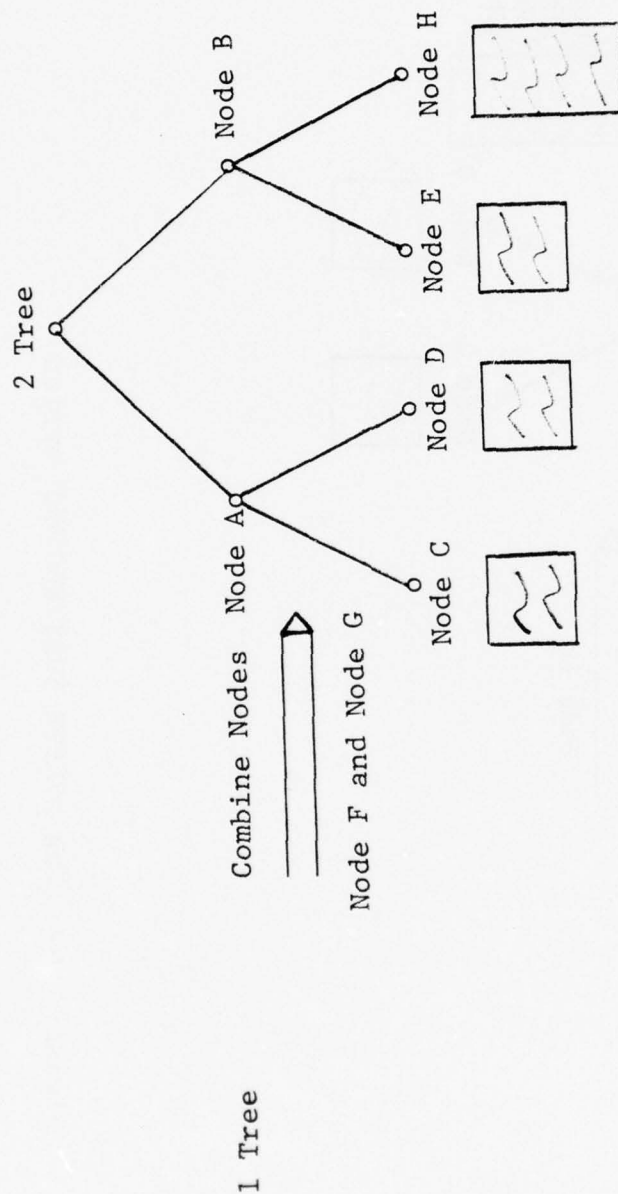


Figure: 7-5 The COMBINE NODES option



## 7.8 APPEND NODES

APPEND NODE is used, as the name suggests, to append a node to a tree. This option is employed when new data become available and the user decides to analyze these data along with data existing in a certain tree. The new data can then be "appended" to the existing tree. The user names the tree and the node to which the new low node has been named, the new node is combined with that low node. In case of vector data, all pertinent mean and covariance files are updated. Figure 7-6 explains this option.

## 7.9 APPEND FEATURES

Extracting the best features from the waveforms for the purpose of pattern analysis and recognition can be a time-consuming affair requiring a lot of imagination and insight on the part of the analyst. Thus, it is unlikely that he will be able to define the best feature vector in one computer session. The APPEND FEATURE option enables the analyst to combine the features extracted at different times into a one feature vector. Say, for example, that the analyst identifies three features from a given waveform data tree and creates a feature vector data tree with vectors of dimension 3. Several sessions later, he identifies two more features from the same waveforms that can aid classification. He creates another data tree with vectors of dimension 2. Finally, with the APPEND FEATURE option, he can combine the two vector data trees and create a vector data tree with vectors of dimension  $(3+2)=5$ . Figure 7-7 explains this option.

## 7.10 SORT

It has been mentioned earlier that each waveform in WPS has a temporary class symbol associated with it. The class symbol is attached at the time of digitization or via an option in the EDIT frame. Given a waveform data tree, the SORT option creates another tree with a senior node and only one level of several low nodes; each low node contains waveforms with the same class symbol. Of the six characters in the new node names, the first is the tree symbol (as supplied by the user), the next four are the letters N, O, D and E, and the sixth is the relevant class symbol; e.g. TNODEA. Figure 7-8 explains this option.

## 7.11. CREATE TREE

This option allows a user to create a tree from a set of lowest or senior nodes from as many trees as the names of which may be entered on one VG line. The created tree may have only one level of low nodes. The low nodes from different levels of a tree can be combined, along with nodes from other trees, into a low node of the new tree. Figure 7-9 explains this option.

## 7.12. CREATE DATA SET

In designing logics for pattern classification, it is customary to split the total data set into a "design" set and a "test" set. Typically, the

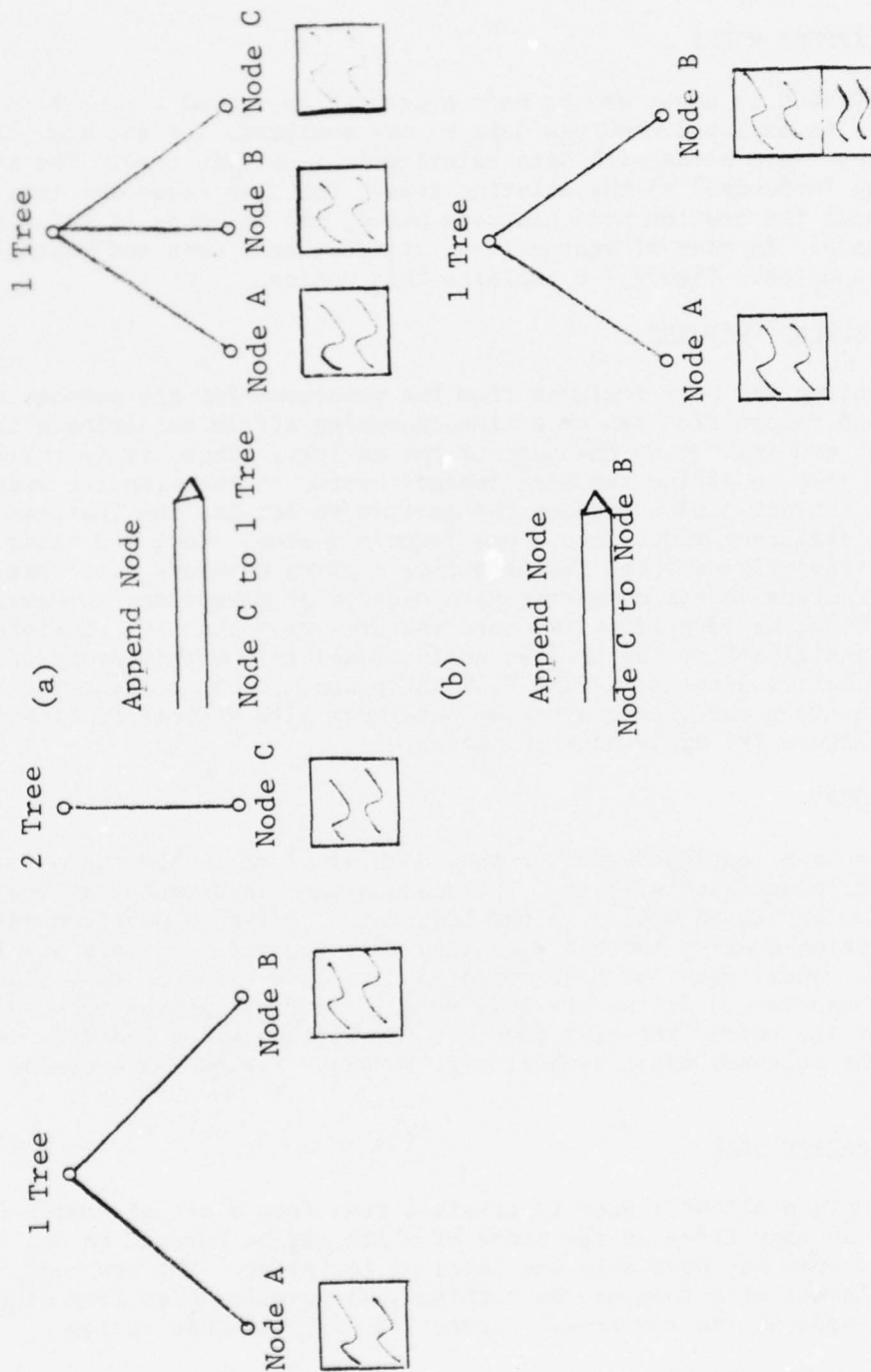


Figure: 7-6 The APPEND NODES option

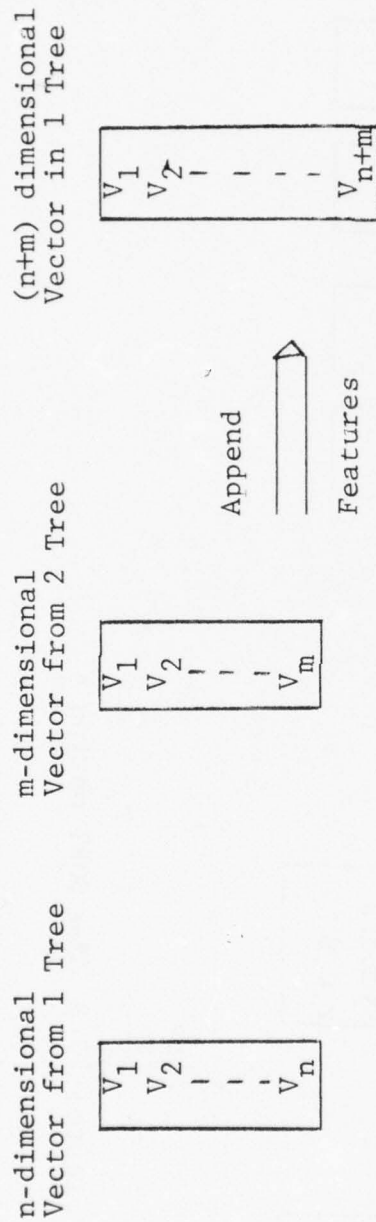


Figure: 7-7 The APPEND FEATURES option

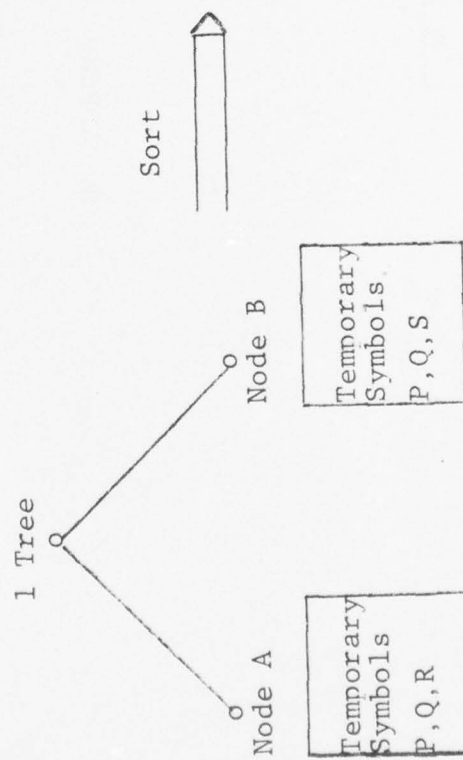
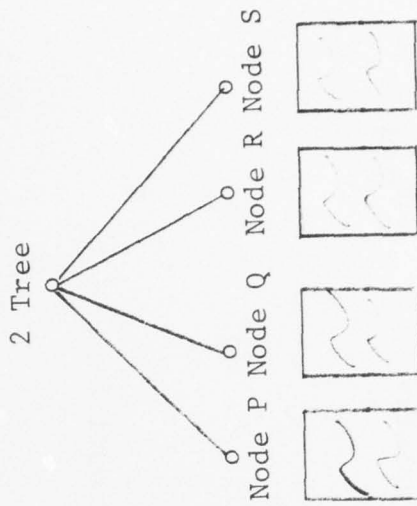


Figure: 7-8 The SORT option



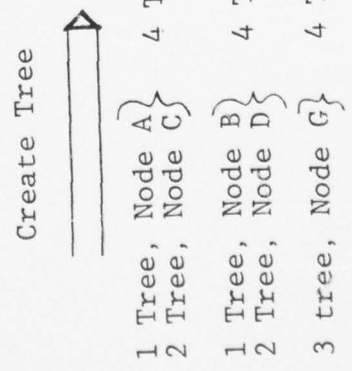
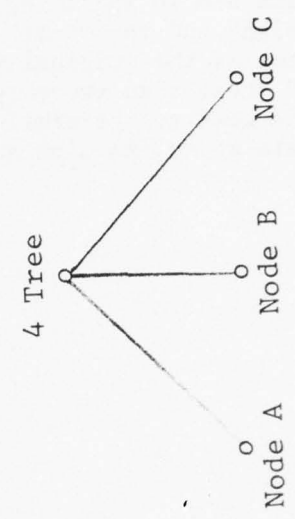
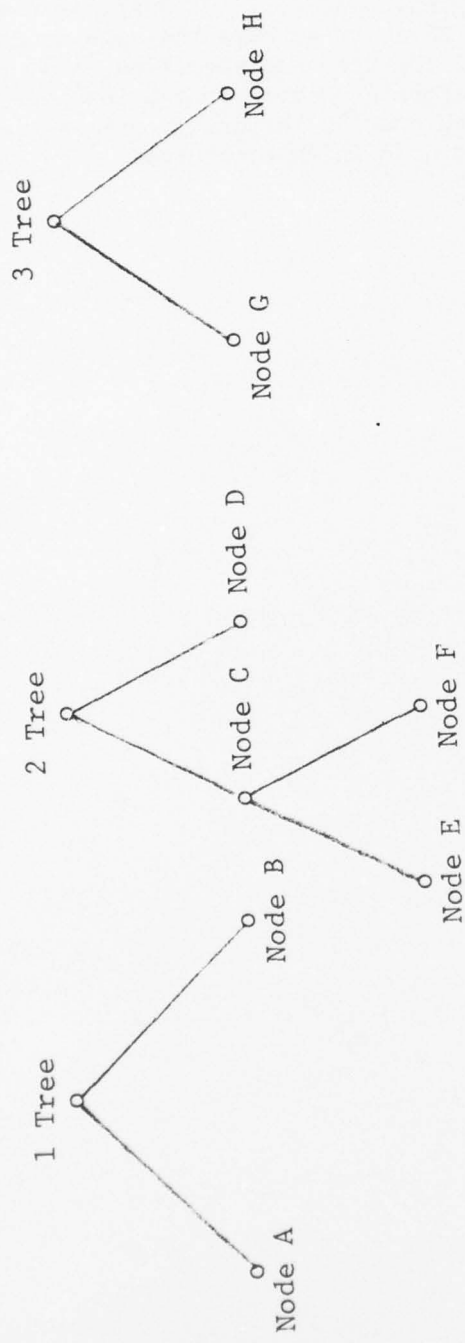


Figure: 7-9 Example of the CREATE TREE option

analyst uses 10% of the available data for structure analysis and logic design and then evaluates the logic on the remaining 90% of his data. The CREATE DATA SET helps him in splitting the original data set. He supplies the names for the "design" and the "test" trees. These trees have the same structure and node-names as the original tree. If the user has specified, say, 10% for the "design" tree, then every tenth waveform or vector from a low node of the original tree goes to the similarly named node of the design tree while the remaining data go to the similarly named node in the test tree.

## SECTION 8

### INPUT/OUTPUT FRAME

The WPS is an interactive system and the emphasis is on "interaction" between the analyst and the computer. However, many analyses call for off-machine time also, when the analyst can concentrate on more involved approaches to the pattern classification problem. For situations like this, the analyst needs a hardcopy of the data stored in the system. The options under the OUTPUT frame allow the data to be printed on paper via the line printer. The printed data come in handy during the interactive session also because these data are not displayed on the VG in an equally convenient form.

The OUTPUT frame also contains options that allow the saving of data on magtapes. Correspondingly, the INPUT frame contains options that allow the re-introduction of such data from magtape and also from cards. Table 8-1 contains the options in the INPUT and OUTPUT frames.

Following is a brief description of these options.

#### 8.1. OUTPUT WAVE TAPE, INPUT WAVE DATA

The first option saves one or more waveform data trees on a 9-track magtape, in the specified WPS tape format. The second option reads the data off the tape and creates a waveform data tree. When the data are stored on the tape, the structure of the original tree is not preserved. When re-creating the tree, the user can, thus, create only one level of low nodes. New tree and node names may be assigned. It is not necessary to read in all the nodes on the tape. However, the nodes cannot be skipped since only sequential reading is allowed.

#### 8.2. OUTPUT VECTOR TAPE, VECTOR TAPE INPUT

This pair of options is similar to the above pair of options, except that they apply to vector data.

#### 8.3. WAVE CARD INPUT

Waveform data may be introduced to the system via cards. The main purpose is the creation of a prototype waveform for some experimental waveform manipulations, but an entire data set may also be constructed. Both time domain and spectral domain data can be entered via this option.

#### 8.4. VECTOR CARD INPUT

Similar to the above option, except that vector data are entered via this option. A complete vector data tree may be entered. The system computes the necessary mean and covariance files.

INPUT DATA FRAME		OUTPUT DATA FRAME	
00	WAVEFORM TAPE INPUT		PRINT DIRECTORY
01	VECTOR TAPE INPUT		PRINT TREES
02	WAVE CARD INPUT		PRINT TR TABLES
03	VECTOR CARD INPUT		PRINT HEADER INFO
04	RESTORE WAVE TREE		PRINT WAVE
05	RESTORE VECTOR TREE		PRINT VEC
06	NAME AND SAVE		PRINT TREE/NODE
07	MODIFY TREE TEXT		PRINT IDS
08	INPUT LOGIC TREE		OUTPUT WAVE TAPE
09	Blank		OUTPUT VEC TAPE
10	"		SAVE TR ON TAPE
11	LIST DIRECTORY		SAVE LOGIC TREE
12	LIST TREES		LIST TREES .
13	DRAW TREES		SELECT DATA SET
14	RETURN TO INIT FRAME		DRAW TREES
15	HARDCOPY		RET TO

Table 8-1 Menu of Options in the INPUT DATA and OUTPUT DATA Frames



#### 8.5 SAVE TR ON TAPE, RESTORE WAVE TREE

The first option saves a waveform data tree on 9-track magtape and the second option reads it back. An exact disk copy of the tree is saved on the tape; therefore, when restoring the tree, the user only needs to mount the appropriate tape on the magtape unit and select the RESTORE WAVE TREE option. No dialogue is necessary and the original tree structure is reproduced.

#### 8.6 SAVE LOGIC TREES, INPUT LOGIC TREES

This pair of options is similar to the above pair but applied to logic trees. Also, more than one logic tree can be saved on a tape, and any or all of them can be read back to the system. Since trees can be read in only by their names, the user is expected to have a record of the trees saved on the tape.

The structure of a logic tree cannot be meaningfully altered. Therefore, the disk-tape-disk transfer maintains the structure. In fact, the restored tree is an exact copy of the original tree.

#### 8.7 PRINT TREES

This option is similar to the LIST TREES option included in many frames. The PRINT TREES option lists the tree names and the associated texts on the line printer, while the LIST TREES does the same on the VG.

#### 8.8. PRINT TREE TABLE

This is equivalent to the DRAW TREE option, which draws the tree structure on the VG. The PRINT TREE TABLE option prints the tree structure schematically on the line printer. Nodes of the same level are printed directly beneath each other. Nodes of a lower level are printed with a greater indentation from the left margin.

#### 8.9. PRINT HEADER INFO

The waveform header information is printed on the line printer. A serial number keeps track of the number of waveforms outputted.

#### 8.10. PRINT WAVEFORM

The numerical values of the waveforms are printed on the line printer. The waveforms as stored on the system have a scale factor which is a power of two. To make it convenient for the analyst, the data are printed with an "exponent" which is a power of ten. The desired waveform is specified by its ID number. The waveform text is also printed out.

#### 8.11. PRINT VECTOR

This option prints out the vectors in a manner similar to the printout of the waveforms.

8.12. PRINT TREE/NODE

With this option, the user can have all the waveforms or vectors in one or more nodes, or the whole tree, printed out. A serial number is attached to each waveform or vector and these are grouped under the node names. The header information is also printed along with the data.

8.13. PRINT IDS

This option outputs to the line printer all the ID numbers under specified nodes or for a whole tree.

## SECTION 9

### ALGEBRAIC/CALCULUS and ARITHMETIC FRAME

Section 5 and 6 described the waveform display options in WPS. The display capabilities are an important part of WPS because WPS is an interactive waveform processing system that emphasizes the graphics.

However, along with the display and editing capabilities, mathematical tools are also necessary to "process" waveforms. A large number of mathematical operations are available as preprogrammed options in WPS. The calculus and arithmetic type operations are included in the two frames described in this section; the options in these two frames are listed in Table 9-1.

Except for the first four options in the ARITHMETIC frame, all other options operate on single waveforms and produce single waveforms. The first four options in the arithmetic operations have single waveforms as outputs but may have two waveforms as inputs. The rest of this section describes the major options in these two frames.

#### 9.1. NORMALIZE

For one reason or another, it is often necessary to normalize a group of waveforms before further analysis can be done. With the NORMALIZE option, the waveforms may be normalized in one of two ways.

##### Magnitude Normalization

In this method, the waveforms are either (1) rescaled individually such that the magnitude of the maximum peak of each waveform equals unity or (2) scaled as a group such that the magnitude of the maximum peak of the group equals unity. In the first method, each waveform is divided by the magnitude of its max peak and in the second, all the waveforms in the group are divided by the magnitude of the overall max peak.

##### Energy Normalization

Under this option the user may rescale the waveforms such that either (1) each waveform has unit energy or (2) the overall maximum of the energy contents is unity.

#### 9.2. SMOOTHING

The raw waveform data are likely to be contaminated with additive noise and therefore smoothing will probably be necessary. Designing filters for optimum smoothing is an art in itself and may be accomplished elsewhere in the WPS system. Under this particular option however, only a "moving average" type smoothing will be provided. The user will have control over the size of the window and the amount by which it is moved each time. It should be recalled that the "moving average" operation is equivalent to low-pass filtering.

ALGEBRAIC/CALCULUS FRAME		ARITHMETIC FRAME
00	NORMALIZE	ADD/SUB
01	SMOOTHING	MULT/DIV
02	ARITHMETIC	$Y = Ax + B$
03	DEMODULATION	$Y = A(x\text{-WAVE})$
04	RECTIFICATION	EVERY PTH PT
05	INDEFINITE INTEGRAL	SHIFT
06	DIFFERENCE	LATERAL GRIDS
07	EXPONENTIAL	DC REMOVAL
08	LOG	Blank
09	<u>+</u> LIMITER	"
10	LIST TREES	SPEC PROTOTYPE
11	SELECT DATA SET	LIST TREES
12	DRAW TREE	DRAW TREES
13	RETURN TO WAVE TO WAVE	SEL DATA SET
14	RETURN TO WPS FRAME	RET TO AL/CAL
15	xxx	RET TO W/W TRANS

Table 9-1 Menu of options for the ALGEBRAIC/CALCULUS and the ARITHMETIC Frames.



More specifically, the "moving average" operation is a convolution in the "time" domain and therefore a product in the "frequency" domain. The rectangular window has a low-pass (sinc) characteristic in the frequency domain. The product in the frequency domain thus causes the reduction of the higher frequency components of the original signal. The width of the first peak of this low-pass filter is controlled by the width of the window in the time domain.

### 9.3. DEMODULATION

Sometimes, a desired signal is available only as an amplitude or a frequency modulation of a carrier wave. In most cases, the waveform can be passed through demodulating circuits before being digitized. (High-quality analog recorders store waveforms as frequency modulated carriers to provide for higher dynamic range of the signal than the tape would normally allow. However, the FM tape is usually played back on compatible hardware, so that the modulation is removed, and made transparent to the user).

In case the digitized waveform is a modulated version of a carrier, provision has been made for the demodulation of such waveforms.

#### Amplitude Demodulation

The method essentially consists of full wave rectification followed by the elimination of all but the peak points. The sampling rate (SR) gets altered due to the elimination of points. The SR may be restored to the original value by means of interpolation between the peak points. Optionally, the DC value may be removed.

#### Frequency Demodulation

The frequency demodulation is done by counting the number of zero crossings in a moving window. The user chooses the size of the window. The size of the window may be chosen from a knowledge of the center frequency and the degree of modulation, or it may have to be chosen by trial and error.

### 9.4. RECTIFICATION

With rectification, all negative-valued points of a waveform can either (1) be eliminated (half-wave) or (2) be replaced by points with equal but positive-valued points, i.e. all negative signs eliminated (full-wave).

### 9.5. INDEFINITE INTEGRAL

The discrete equivalent of the indefinite integral is the running sum of the points in the waveform.

### 9.6. DIFFERENCE

The first and second difference operations are the discrete equivalents of the first and second derivatives. The first difference is the difference

between consecutive points of a waveform. The second difference is the "first difference" operation applied to the first difference.

#### 9.7. EXPONENTIAL

With this option the square and cube of the waveform may be obtained.

#### 9.8. LOG

The  $\log_2$  or  $\log_{10}$  may be computed with this waveform.

#### 9.9. + LIMITER

The user selects the upper and lower bounds of the limiter. All points of the waveform exceeding these bounds are clipped to the boundary values. The effect is very similar to that of a hardware limiter.

#### 9.10. ADD/SUB

A scalar or prototype waveform may be added to or subtracted from the waveforms in a tree. The ability to subtract a waveform from another facilitates comparison of waveforms.

#### 9.11. MULT/DIV

The waveform may be multiplied or divided by a scalar or a prototype waveform. The ability to multiply with a waveform furnishes the capability of applying a time-varying scale factor to counteract, for example, a time-varying gain which was encountered during recording of the signal.

#### 9.12. $y = Ax + B$

In this operation, A and B are scalars and x stands for the waveform. The option combines the multiplication and addition. There is no restriction on either the magnitude or the signs of the scalars A and B.

#### 9.13. $Y = A(x\text{-WAVE})$

Here, A is a scalar, x is the signal waveform and "WAVE" is a designated prototype waveform.

#### 9.14. EVERY PTH POINT

This option is to be used to reduce the effective sampling rate of over-sampled waveforms by retaining every  $p^{\text{th}}$  point of the waveforms.

#### 9.15. SHIFT

To compare two waveforms meaningfully, it is usually necessary that the sampling instants be the same. If the sampling instants are not the same,

it may be necessary to shift one of the waveforms forward or backward in time, by a fraction of the sampling interval. At other times, it may be necessary to introduce a shift in the frequency domain, as in heterodyning.

#### Time Shift

Multiplying a waveform by a factor  $e^{-jW(\Delta T)}$  in the frequency domain causes the time domain waveform to be shifted forward in time by an amount  $\Delta T$ . Thus, to obtain a time shift, one takes the FFT of the waveform, multiplies it by the exponential factor, and then computes the inverse FFT of the product.

#### Frequency Shift

Multiplying a waveform by a factor  $e^{+j(\Delta W)T}$  in the time domain causes the frequency domain waveform to be shifted forward in frequency by an amount  $\Delta W$ . Thus, a frequency shift is performed by multiplying the time domain waveform by the exponential factor.

#### 9.16. LATERAL GRIDS

This option provides an amplitude histogram of the waveforms. The user selects certain grid levels which then become the amplitude "bin" sizes. The histogram is a plot of the percentage of time the waveform was within each bin.

#### 9.17. DC REMOVAL

In most cases, the average or the DC value of a waveform carries no information, and at the same time distorts displays or injects large numbers into computation. Under this option, the average of each waveform in a tree is computed and subtracted from the respective waveforms.

## SECTION 10

### SPECTRAL ANALYSIS MODULE

This module contains all the WPS options for transforming time domain waveforms into frequency and quefrency domains and vice versa. The output options are listed in Table 10-1. The basis for all transformations in this module is the floating point, Fast Fourier Transform (FFT).

The input to the module may be an entire tree, a node from a tree, a single waveform, or a part of a waveform. The waveform currently displayed in the Single Waveform Display module may be designated as an input. For any transform, a "window" from the waveform is taken as the input and an output waveform (or two waveforms, if the output is complex) is produced. The window can be made to slide over the input waveform at a specified rate. The window size (i.e. number of points) has to be a power of 2, because the particular FFT routine employed does not work for any other type of window. Also, the window can be no smaller than eight points ( $2^3$ ) and no larger than 1024 points ( $2^{10}$ ), with the current version of the FFT routine.

The transforms can accept real or complex data. There is no special way of storing imaginary data. The imaginary waveforms are stored just like real waveforms. It is up to the analyst to supply the names of the two trees containing the real and imaginary parts of the data. The structure of the two trees should be same and all similar nodes should have the same names.

The user has a choice of applying a rectangular (default), Hanning, Hamming, or a Blackman weighting to each input window.

The mathematical definitions of the various output options are as follows. Let  $f(t)$  be the real or complex, time domain waveform. Then (See Digital Processing of Signals by Gold & Rader, McGraw-Hill, 1969):

#### 1. FFT

$$F(k \Omega) = \sum_{n=0}^{N-1} f(nT) e^{-j \Omega T n} \quad 0 \leq n, k \leq N-1$$

where

$$\Omega = \frac{2 \pi}{NT}$$

$$N = \text{Window size, } 8 \leq N \leq 1024$$



00	INPUT DATA SET
01	OUTPUTS
02	WAVE SEQUENCE #
03	POINT SEQUENCE #
04	WINDOW OPTIONS
05	WEIGHTING FUNCTIONS
06	WINDOW FILL
07	CHECK SPECTRAL PARAMETERS
08	EXECUTE
09	Blank
10	LIST TREES
11	WAVE/WAVE TRANSFORMATIONS
12	SINGLE WAVE/EDIT
13	MULTIPLE WAVE DISPLAY
14	RETURN TO WPS FRAME
15	HARDCOPY

TABLE 10-1 MENU OF OPTIONS IN THE SPECTRAL ANALYSIS MODULE

AD-A034 417

PATTERN ANALYSIS AND RECOGNITION CORP ROME N Y  
THE WAVEFORM PROCESSING SYSTEM (WPS). VOLUME I. (U)  
OCT 76 P K SANYAL, D H FOLEY

F/6 9/2

UNCLASSIFIED

PAR-76-12

RADC-TR-76-224-VOL-1

F30602-72-C-0193

NL

2 OF 2

AD  
A034417



END

DATE  
FILMED  
2-77

$T$  = Sampling interval of  $f(t)$

$F(k \Omega)$  is a complex waveform.

## 2. FFT Magnitude

This is the magnitude of  $F(k \Omega)$  and is real.

$$\text{Magnitude} = \sqrt{(\text{Real})^2 + (\text{Imaginary})^2}$$

## 3. FFT Phase

This is the phase of  $F(k \Omega)$  and is real. The phase lies between  $\pm\pi$ .

## 4. Log FFT Magnitude

This is the natural logarithm of the magnitude of the FFT. If there are zero points, the logarithm cannot be computed. Therefore, all zero points are replaced by a small non-zero, positive  $\epsilon$ . Whenever an output involving a logarithm is requested, the user is asked to supply a value for  $\epsilon$ . Any positive value may be chosen. A unity value is often suitable, since the log of unity is zero. The choice of  $\epsilon$  usually depends upon the range of numbers in the waveform; i.e.,  $\epsilon$  should be chosen so that it should be possible to tell easily from the log plot which points were replaced by  $\epsilon$ . The default value for  $\epsilon$  is  $1 \times e^{-38}$ , which is approximately the smallest non-zero floating point number on the PDP-11/45 computer.

## 5. Inverse FFT

The inverse FFT converts data from the frequency domain to the time domain. The equation is

$$f(nt) = \sum_{k=0}^{N-1} F(k \Omega) e^{j \Omega T n k} \quad 0 \leq n, k \leq N-1$$

In general, the output is complex.

## 6. Power

This is the square of the magnitude of the FFT.

## 7. Log Power

This gives the natural logarithm of Power. All zero points are replaced with an  $\epsilon$ .

## 8. Cepstrum

The cepstrum of a time domain waveform is defined as the

$$\text{Power} \{ \log_e [\text{Power} \{ f(t) \} ] \}$$

This transform had been suggested by researchers in the area of speech analysis. The purpose of the transform is to distinguish two waveforms that are convolved in the time domain. The premise for the transform is that two waveforms that are convolved in the time domain are multiplied in the frequency domain. The logarithm converts the product into a summation. Finally, the Power transform separates the two additive components. (See "Digital Signal Processing" by Oppenheim & Shafer, Prentice Hall, 1975).

The Cepstrum is an irreversible transform. The Cepstrum is meant for real data and produces real data.

## 9. Complex Cepstrum

The Complex Cepstrum is defined as

$$\text{FFT}^{-1} \{ \log_e [\text{FFT} \{ f(t) \} ] \}$$



and the purpose is the same as the Cepstrum. The Complex Cepstrum is usually applied to real data (the user is not restricted from applying it to complex data, though some of the computations are based on the assumption that the input is real) and the output is, accordingly, real. (See "Digital Signal Processing" by Oppenheim & Shafer.)

The logarithm in the expression for complex cepstrum is a complex log because the argument is complex. This is so because the FFT of a waveform is, in general, complex and can be expressed as  $a e^{j\theta}$  where "a" is the magnitude and "θ" the phase of the complex quantity. The complex logarithm produces the sum  $\{(\log a) + j\theta\}$ . When computing the complex log, the phase "θ" has to be expressed as a continuous quantity and not just the principal value lying within  $\pm \pi$ .

The complex cepstrum, unlike the cepstrum, is a reversible transform.

#### 10. Continuous FFT Phase

In general, the FFT is a complex waveform, consisting of the real and imaginary parts. This is also expressed in terms of the magnitude and a phase. Typically, the phase is computed as the principal value of an arctan function. The principal values lie within  $\pm \pi$ . A continuous phase can be computed based on the assumption that the phase waveform is odd symmetric about the foldover frequency.

#### 11. Inverse Cepstrum

The complex cepstrum is reversible and the inverse cepstrum is defined as

$$\text{FFT}^{-1} \{ \text{Exp} [\text{FFT} \{ f(t) \} ] \}$$

The Inverse Filter Coefficients, Reflection Coefficients, and the Formant Spectrum were specifically programmed for speech analysis and the reader is referred to the short course entitled "Linear Prediction of Speech - Theory and Practice" - by J.D. Markel, A.H. Gray Jr., and H. Wakita of Speech Communications Research Laboratory, Inc. 1973.

The Inverse Filter essentially attempts to transform the input signal into a constant or white noise spectrum; i.e., the Inverse Filter is approximately the inverse of the signal characteristics. The Reflection Coefficients are quantities necessary for the computation of the inverse filter parameters. The Formant Spectrum gives the pitch period of the speech data being analyzed.

For any input data, the user can select any, several or all of the output options. In addition, there are options that allow the user to massage the input data stream in several ways. The options in the spectral Analysis Module are shown in Table 10-1, page 10-2.

THE DEFAULT OR INITIAL WPS PARAMETERS USED IN THIS MODULE ARE AS FOLLOWS:

INPUT IS CURRENT DATA SET (ASSUMES REAL)  
OUTPUT IS POWER SPECTRUM

WAVE SEQUENCE \* IS 1

Ø IMPLIES CURRENT WAVEFORM ONLY

POINT SEQUENCE \* IS 1

WINDOW: 1ST M POINTS OF EACH WAVEFORM (M=1024)

WEIGHTING FUNCTION: HAMMING FOR INVERSE FILTER. NONE FOR THE REST.

LET L BE NUMBER OF POINTS IN WAVEFORM.

IF  $L < M$ , THEN LAST  $M-L$  POINTS FILLED WITH AVERAGE VALUE.

IF  $L < M$ , THEN LAST  $L-M$  POINTS IGNORED.

OUTPUT TREE TEXT IS DUPLICATED FROM INPUT TREE

D.C. LEVEL ON INPUT SIGNAL IS REMOVED

YOU MAY OVERRIDE ANY OF THESE PARAMETERS BY SELECTING OPTIONS ON LEFT.

FIGURE 10-1 THE COMBINATION OF DEFAULT OPTIONS IN THE SPECTRAL ANALYSIS MODULE

1. FFT
2. FFT MAGNITUDE
3. FFT PHASE
4. LOG FFT MAGNITUDE
5. INVERSE FFT
6. POWER
7. LOG POWER
8. CEPSTRUM
9. COMPLEX CEPSTRUM
10. CONTINUOUS FFT PHASE
11. INVERSE CEPSTRUM
12. INVERSE FILTER COEFFICIENTS
13. REFLECTION COEFFICIENTS
14. FORMANT SPECTRUM

TABLE 10-2 OUTPUT OPTIONS IN THE SPECTRAL ANALYSIS MODULE

When the Spectral Analysis Module is selected for the first time, a combination of default options is displayed on the VG screen. This combination is shown in Figure 10-1. If the default parameters are satisfactory, then the user may select the EXECUTE option and obtain the Power transform of the current data set. Any of the default parameters may be changed by selecting the appropriate option from the menu.

#### 10.1. INPUT DATA SET

If other than the current data set is desired, then the user should exercise this option. (The "current data set" is selected before the user gets to the Spectral Analysis Module, typically in the Single Display Frame).

#### 10.2. OUTPUTS

The list of output options is shown in Table 10-2. When the option OUTPUTS is selected by the user, this table is displayed on the VG and the user selects the options by typing in the corresponding number. One, several, or all of the output options may be selected.

#### 10.3. WAVE SEQUENCE #

The user may elect to process each waveform or skip some of them by specifying a sequence #. He may also select the waveform last displayed in the Single Waveform display frame by specifying a wave sequence # of 0.

#### 10.4. POINT SEQUENCE

This option allows the user to reduce the effective sampling rate of the input waveform. If  $P=3$ , then the effective sampling rate is reduced by 3. The original waveform is not disturbed.

#### 10.5. WINDOW OPTIONS

Only one window or several windows from each waveform may be processed. If only one window is to be processed, the user specifies the window size and the starting point. If desired, the start point can be the "begin marker" stored in the header of the waveform.

For sliding windows, the user additionally specifies the amount of slide. He also specifies the number of windows to be processed from each waveform. Alternatively, he may indicate that as many windows as possible should be processed.

#### 10.6. WEIGHTING OPTIONS

It is known that the windowing operation distorts the frequency domain representation of the waveform because the "windowing" is actually multiplying the original waveform with a rectangular waveform (window). This results in a convolution of the Fourier transform of the original waveform with the



Fourier transform of the rectangular waveform. Due to the side lobes of the rectangular waveform, the result of the convolution can be significantly different from the desired actual frequency domain representation of the original waveform.

The problem with sidelobes can be reduced by applying a "weighting function" to the window (See Oppenheim and Schafer). Different people have suggested different weighting functions and not one can be uniquely recommended for all purposes. Therefore, three weighting functions, in addition to the rectangular (or no) weighting, are available, namely

$$(a) \text{ Hanning} \quad w(n) = \frac{1}{2} [1 - \cos \left( \frac{2\pi n}{N-1} \right)] \quad 0 \leq n \leq N-1$$

$$(b) \text{ Hamming} \quad w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{N-1} \right) \quad 0 \leq n \leq N-1$$

and

$$(c) \text{ Blackman} \quad w(n) = 0.42 - 0.5 \cos \left( \frac{2\pi n}{N-1} \right) + 0.08 \cos \left( \frac{4\pi n}{N-1} \right) \\ 0 \leq n \leq N-1$$

It is left to the analyst to determine which weighting is the best for his purpose.

#### 10.7. WINDOW FILL

The FFT routine can process only windows whose size is a power of 2 and therefore the user is compelled to specify such a window size. Even then, there can be waveforms that do not have enough points to fill a window. In such cases, the system will automatically insert zero magnitude points to fill a specified window. All the zeros may be inserted at the end or they may be inserted half at the front and half at the end, as the user wishes.

## SECTION 11

### SEGMENTATION FRAME

When a signal is recorded, it is not always possible to register each event or complex as a separate waveform. On the other hand, for the purpose of analysis, it is necessary to separate each event into a separate waveform. The user can do this manually in the EDIT frame, where he can employ the TRUNCATE FRONT, TRUNCATE END, and SEGMENT WAVE options over and over to create waveforms containing single events only. If there are many waveforms to be created, the manual process can be tedious.

The Segmentation Frame contains options that can be used to automatically segment waveforms and create new waveforms containing signals of interest only.

For automatic segmentation, the system needs a means to recognize the beginning and the end of a desired signal. The procedure is as follows.

Take a "window" of the given waveform and compute a specified functional. Check the functional against a specified threshold and determine if a specified logic, e.g. =, < etc., is satisfied. If yes, then place a begin marker at the beginning of the window or a specified distance before it and go on to locate the end of the desired signal. If no, then slide the window a specified amount and start the procedure again.

The search for the end of the signal is done in a similar manner, except that the end marker is placed at the end or at a specified distance after the end of the window.

#### 11.1. SEGMENTATION PARAMETERS

The search for begin and end of signal is done alternately; i.e., first a begin marker is located, then an end marker, then again a begin marker and so on. Some of the quantities pertinent to the segmentation are shown in Figure 11-1 on page 11-2, and defined below:

$\tau_b$ : The size of the window for the begin marker.

$T_b$ : The threshold for checking the functional against . (for the begin marker).

$\alpha$ : The advance before the beginning of the window, for placing the begin marker.

$S_b$ : Amount of slide for the window when searching for the begin marker.

$\tau_e$ ,  $T_e$ ,  $\beta$  and  $S_e$  are for the end marker and correspond to  $\tau_b$ ,  $T_b$ ,  $\alpha$  and  $S_b$ .

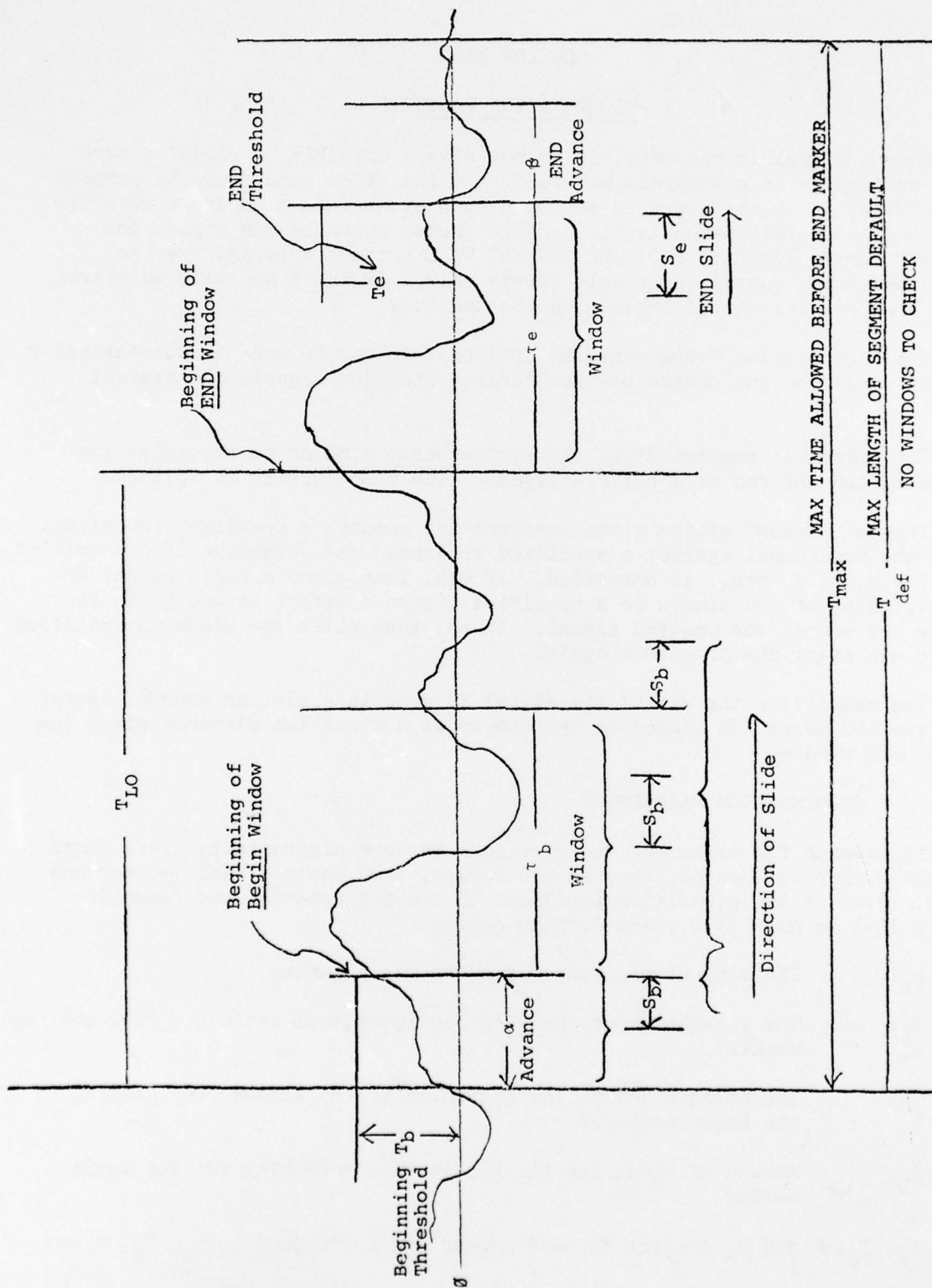


Figure 11-1 Begin and End Parameters



- $T_{lo}$ : The lock-out period. After a begin marker has been located, a certain amount of the following portion of the waveform is skipped before the search for the end marker is initiated. The lock-out period is allowed either because the analyst desires at least a certain length of the segmented waveform or because it is known that the end of the signal cannot be within a certain distance of the begin marker; thus, specifying a lock-out period cuts out some unnecessary search.
- $T_{max}$ : This is the maximum period between the begin marker and the end marker. If the end marker criterion fails to locate an end marker before this distance, then the criterion is ignored and a marker is placed at the distance  $T_{max}$ . The end marker criterion is activated again after the next begin marker is located.
- $T_{def}$ : When a  $T_{def}$  (default period) is specified, no other criterion is necessary for the end marker. The end marker is placed at a distance  $T_{def}$  from the begin marker.

## 11.2. SEGMENTATION CRITERIA

The analyst has a choice of criteria that he may prescribe for the begin and end markers.

### 11.2.1. Cross Correlation

The cross correlation between a prototype and the window is computed and checked against a threshold. The idea essentially is to check the resemblance of the window with the specified prototype waveform. Each point of the correlation waveform is compared with the threshold; at least one point should satisfy the logic.

### 11.2.2. Convolution

The convolution between the window and a prototype waveform is computed and the peak of the output is checked against a threshold. If the prototype is the impulse response of a filter, then the convolution produces the output of the filter, where the window is the input.

### 11.2.3. Rise In Time Window

The rate of rise of the waveform can indicate the begin or the end of a signal. The rate of rise that is computed is the average rate of rise, i.e. the difference between the last and first point of the window, divided by the width of the window.

### 11.2.4. Fall In Time Window

This is the negative of the rise in the window.



#### 11.2.5. Amplitude Level

This is simply the amplitude of the waveform.

#### 11.2.6. Average Value In Time

The average value is the sum of all points divided by the width of the window.

#### 11.2.7. Average Power In Time

The average power is the sum of the square of all points in the window divided by the width of the window.

#### 11.2.8. Average A-C Power In Window

The A-C power is computed by first finding the A-C component of the waveform in the window. This is done by subtracting the average value (D.C.) from each point. The average A-C power is the sum of the square of the points in the A-C component, divided by the width of the window.

#### 11.2.9. Zero Crossings

This is a count of the number of zero crossings in the window. The count is an indication of the frequency of the waveform.

### 11.3. THRESHOLD LOGIC

Besides specifying a criterion and a threshold, a threshold logic can also be specified. The choices are as follows:

LT  
LE  
EQ  
GE  
GT  
NE

Suppose the criterion "AMPLITUDE LEVEL" is selected and a threshold T and logic GE is chosen. Then all the points in the window are compared with the threshold T and if any of them is  $\geq T$ , then the criterion is assumed to have been satisfied.

### 11.4. MARKER TREE

The segmentation is done as a two-step process; first the markers are created and then these markers are used to segment the waveform. The positions of the markers are stored in a manner very similar to the storing of waveform data (see Section 4). The marker tree has the same structure as the waveform tree from which it was created. The node names and ID numbers are identical and tree names may also be the same, since the data types of the waveform data and the marker data are different.

MARKER TREE

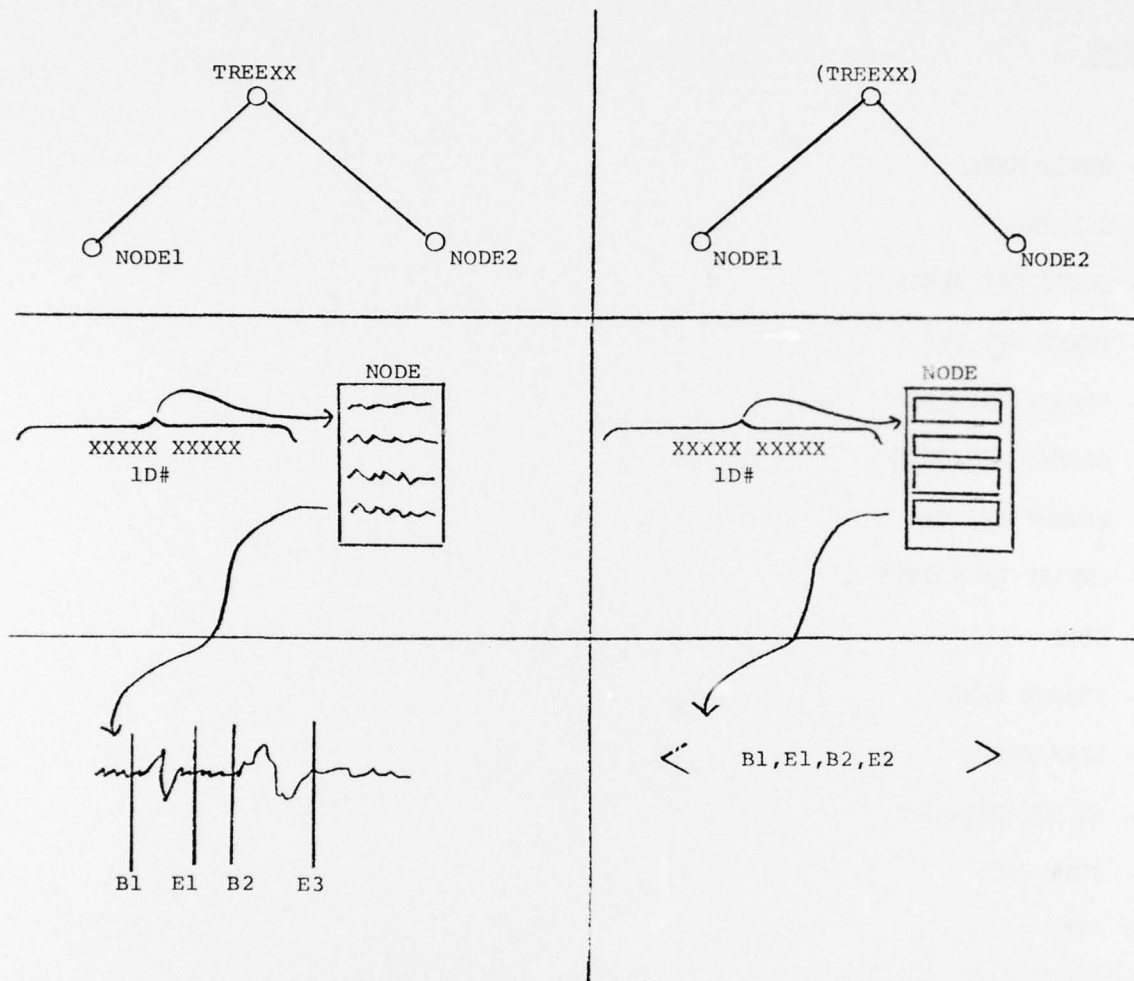


Figure 11-2 Creation Of Marker Tree From Waveform Data Tree

# OPTIONS

- 00 - BEGIN MARK
- 01 - END MARK
- 02 - THRES OPT BEGIN
- 03 - THRES OPT END
- 04 - CREATE MARKERS
- 05 - SEGMNT ASSIGNED
- 06 - SEGMNT APRIORI
- 07 - CREATE PROTOTYPE
- 08 - EDIT
- 09 - SINGLE WAVE
- 10 - LIST TREES
- 11 - SELECT DATA SET
- 12 - DRAW TREE
- 13 - \*\*\*
- 14 - RET TO WPS FRAME
- 15 - RET TO INIT FRAME

TABLE 11-1 MENU OF OPTIONS IN THE SEGMENTATION FRAME

Figure 11-2 shows the process of creation of the marker tree. After the markers are created, the analyst can have the calculated markers displayed on the corresponding waveforms by selecting the appropriate option in the "SINGLE DISPLAY - EXTENDED OPTIONS" frame. From the display the user can determine if the criteria and parameters prescribed by him have performed satisfactorily. If not, he can create fresh markers by specifying different criteria and parameters.

When the placing of the markers appears to be satisfactory, the user can apply the marker tree to segment the original waveform.

#### 11.5. OPTIONS

The menu of options in the Segmentation frame is given in Table 11-1.

By selecting options 00 and 01, namely BEGIN MARK and END MARK, the user prescribes the criteria and parameters for the two types of markers.

Options 02 and 03, namely TRESH OPT BEGIN and TRESH OPT END, allow the alteration of criteria and parameters. The names stand for Threshold Optimization but they allow the alteration of all criteria and parameters.

Option 04 creates the tree consisting of segment markers and option 05 uses the segment marker tree previously created to actually segment the waveform.

Option 06, SEGMNT APRIORI, uses the markers stored in the header of each waveform to segment the waveform. No parameters or criteria need be prescribed for this case.

Segmentation is a polymorphic operation; i.e., in general, more than one segment is produced from each input waveform. As in other polymorphic operations in WPS, new ID's are automatically generated after the user indicates the digits in the original ID that may be updated.

For two of the criteria, CROSS CORRELATION and CONVOLUTION, discussed earlier in this section, a prototype has to be specified. The user may create a prototype waveform by entering the data through cards, using the option WAVE CARD INPUT in the INPUT DATA frame. Alternatively, the prototype may be created by taking a desired segment of a particular waveform already stored in the system. To do this, markers are appropriately placed via the option INSERT APRIORI SEGMENT MARKER in the EDIT frame and the option CREATE PROTOTYPE in the Segmentation frame is then selected. The prototype exists as a single waveform in a tree.



## SECTION 12

### PLAYBACK FRAME

To facilitate the analysis of speech and other acoustic data, a sound playback system has been added to the WPS facility. The different playback options are included in the Playback frame, shown in Table 12-1. Data that is being played back can be displayed at the same time, provided the waveform can be accommodated on one page of the VG screen. (Of course, the waveform can always be displayed with a reduced sampling rate and, indeed, any waveform can be made to fit in one page). The simultaneous visual and aural presentation of acoustic data is a great help to an analyst interested in the pattern recognition aspect of acoustic data analysis.

For sound playback, digital to analog converters (DAC) are necessary. At the RADC Waveform Processing Facility, an AD-5 Hybrid Computer is connected to the PDP 11/45 Digital Computer. There are 16 DAC's on the AD-5, any of which may be used for the sound playback. It is left to the user to couple the output of the DAC to a loudspeaker/headphone system via a patch board. Tunable, analog, low pass/high pass filters may also be optionally included at the input to the loud-speaker system. The digital hardware limits the playback rate to 32767 Hertz, which allows the playback of over 16 kHz signals. This is adequate for most cases; e.g., telephone lines limit voice signals to about 3 kHz.

#### 12.1.      OPTIONS

The first four options in the Playback Frame allow the selection of the data to be played back.

##### 12.1.1.    Play This Part

This option allows the playback of the exact same data that is displayed on the VG screen.

##### 12.1.2.    Play This Wave

The user may listen to the entire waveform or any segment of the waveform which is currently displayed from the Single Wave Display Module.

##### 12.1.3.    Play A New Wave

Any waveform on the disk may be selected for playback. The waveform is identified by the tree, node, and the wave ID.

##### 12.1.4.    Light Pen Play

A portion of waveform currently displayed may be selected for playback. The desired portion of the displayed waveform may be marked off by the light pen.

- 00 - PLAY THIS PART
- 01 - PLAY THIS WAVE
- 02 - PLAY A NEW WAVE
- 03 - LIGHT PEN PLAY
- 04 - Blank
- 05 - "
- 06 - "
- 07 - "
- 08 - DISPLAY SAME
- 09 - DISPLAY NEXT
- 10 - DISPLAY PREVIOUS
- 11 - DISPLAY NAMED
- 12 - RET TO EXT OPT
- 13 - RET TO SINGLE
- 14 - RET TO WPS FRAME
- 15 - HARDCOPY

TABLE 12-1 MENU OF OPTIONS IN THE PLAYBACK FRAME

## SECTION 13

### CONCLUDING REMARKS AND RECOMMENDATIONS

#### 13.1. CONCLUSIONS

In reviewing the history of pattern recognition, a researcher soon realizes that the vast majority of literature and research centers on attempting to answer the following question: Assuming one has a set of  $m$  features or measurements which contain all the necessary information to discriminate between various classes of interest, how does one partition the  $m$ -space to successfully arrive at class decisions?

The history evolves from the early days of the perceptron-type, adaptive "learning" machines to statistically-derived classification algorithms. The evolution continues from the single, "researcher-x," panacea-type algorithm to large interactive systems containing collections of mathematical and graphical algorithms for both the analysis and classification of vector data. Perhaps the most successful of these large interactive systems is the On-Line Pattern Analysis and Recognition System (OLPARS) (7). Although emphasis has shifted from "the single algorithm to solve all problems approach" to the concept of finding algorithms best suited for the data structures at hand, we are still at a point in pattern recognition history where the primary thrust is on the manipulation of vector data bases.

If our hypothetical reviewer continues his faithful search of the literature, he will undoubtedly face the frustrations of the "meaningless statistics." That is, each new classification algorithm will attempt to prove its superiority via its performance on some limited data base. At this junction the reviewer is faced with a difficult choice. Either he must believe that the myriad of papers claiming ninety percent plus correct recognition rates have led to true solutions in many pattern analysis problems, or he will become a skeptic.

Soon the reviewer will uncover a collection of papers in the literature dealing with statistical traps and overly optimistic quotes of future performance (e.g. see Foley (9) and Toussaint (10)). Although these papers will only reinforce his skepticism, the earnest researcher still has to contend with the difficulty of evaluating the value of proposed pattern recognition systems.

One can only conclude that there will never be any satisfactory method for predicting the future performance of any classification logic if that logic is designed using only calculations and/or statistics based on finite data bases (quite naturally limited by manpower and financial collection constraints).



It is therefore felt that the evolution should now reach a stage where the emphasis is on the problem itself; attention has to be paid to systems for displaying, manipulating, and extracting features from the raw data. Only via intimate involvement in the data collection, physics, characteristics, and idiosyncrasies of the particular problem at hand can a researcher devise features which he can reasonably expect to stand the test of time. This understanding of his problem, coupled with the realization that pattern recognition tools are only an aid, offers the most promising approach to bridging the gap between theory and practice in pattern recognition.

Perhaps it is a tribute to the successs of the many researchers who have contributed to the pattern classification literature that the pattern recognition community is at a point where emphasis can shift from algorithms which manipulate vector data to more problem-dependent applications. In effect, answers to the question asked in the initial paragraph have been found. The classification and logic design portion of the problem appears to have been solved for all practical purposes. Attention must now be devoted to methods for making the assumption in the first paragraph a valid one.

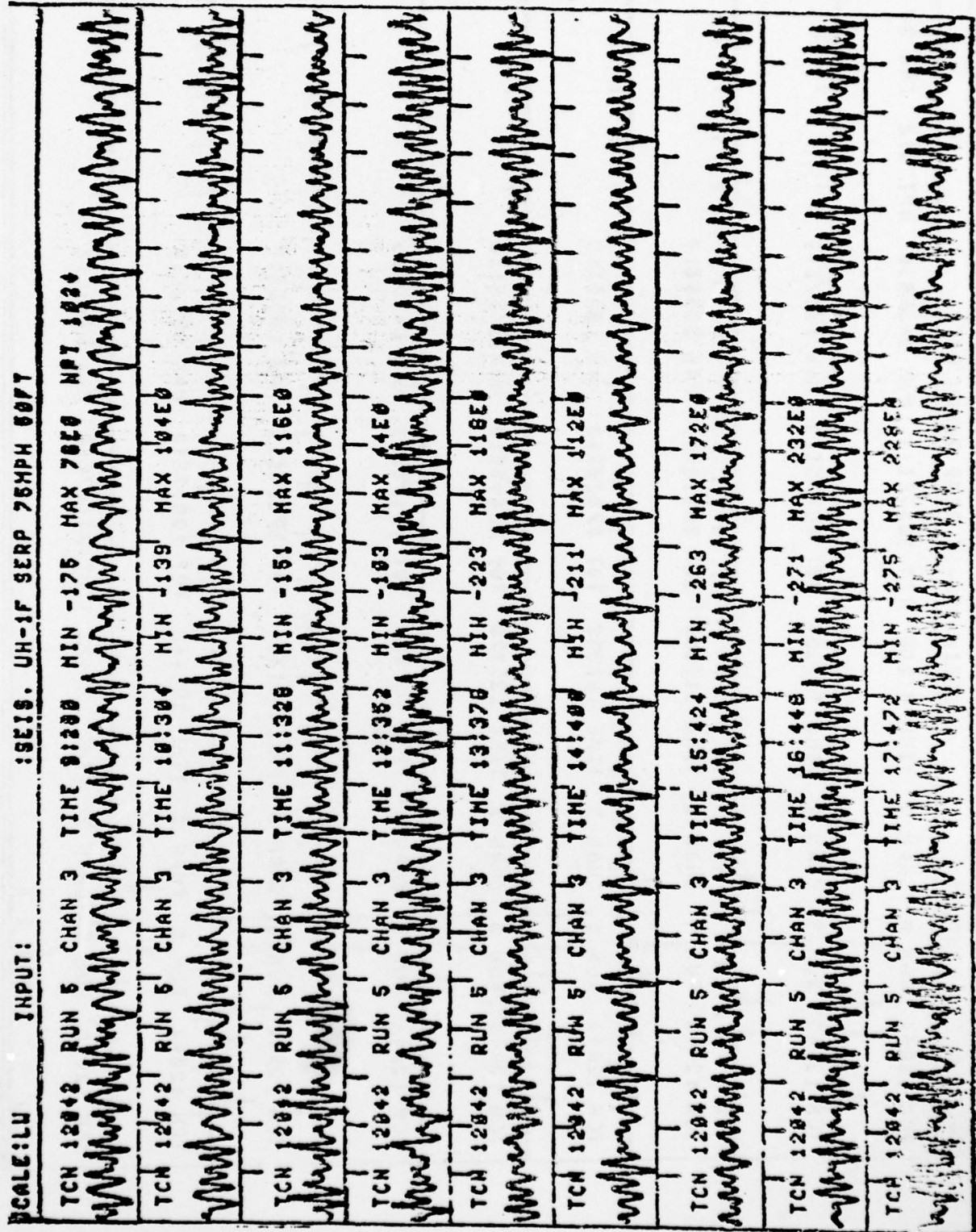
To briefly illustrate the interaction of various display formats, transforms, and the analyst's own knowledge, consider the problem of attempting to recognize the presence of a helicopter from its acoustic signature. All other acoustic sources are to be considered as background, i.e., the nuisance class.

Until data has been collected and displayed, the analyst may only conjecture as to a set of distinguishing or dominant "features" for the class of helicopters. Once data has been carefully collected and displayed, the analyst can attempt to link dominant characteristics with their corresponding target sources. Figure 13-1, page 13-3 shows nine lines of the acoustic signature of a UH-1F helicopter. Each line contains one second of data, and the time at the end of the  $n^{\text{th}}$  line corresponds to the time at the beginning of the  $(n+1)^{\text{th}}$  line. From this display the analyst might hypothesize the periodic nature of the signature and request power spectrums taken over one-second windows.

These spectral plots are shown in Figure 13-2, page 13-4. Now the harmonic nature of helicopter signature is obvious. Via well-designed collection experiments the analyst can link the spacing of the spectral peaks to the blade-passage-frequency of the main rotor. One can easily see that the presence or absence of the harmonic structure and its associated fundamental frequency are the key features for identifying helicopters. It is not the values of any or all of the 1024 power spectral coefficients! It is not the eigenvectors of the lumped data covariance matrix, etc.!

Also note that a skilled analyst can bring to bear the a priori knowledge that the location of spectral peaks changes with the doppler shift phenomena and with helicopter design parameters such as number of blades and





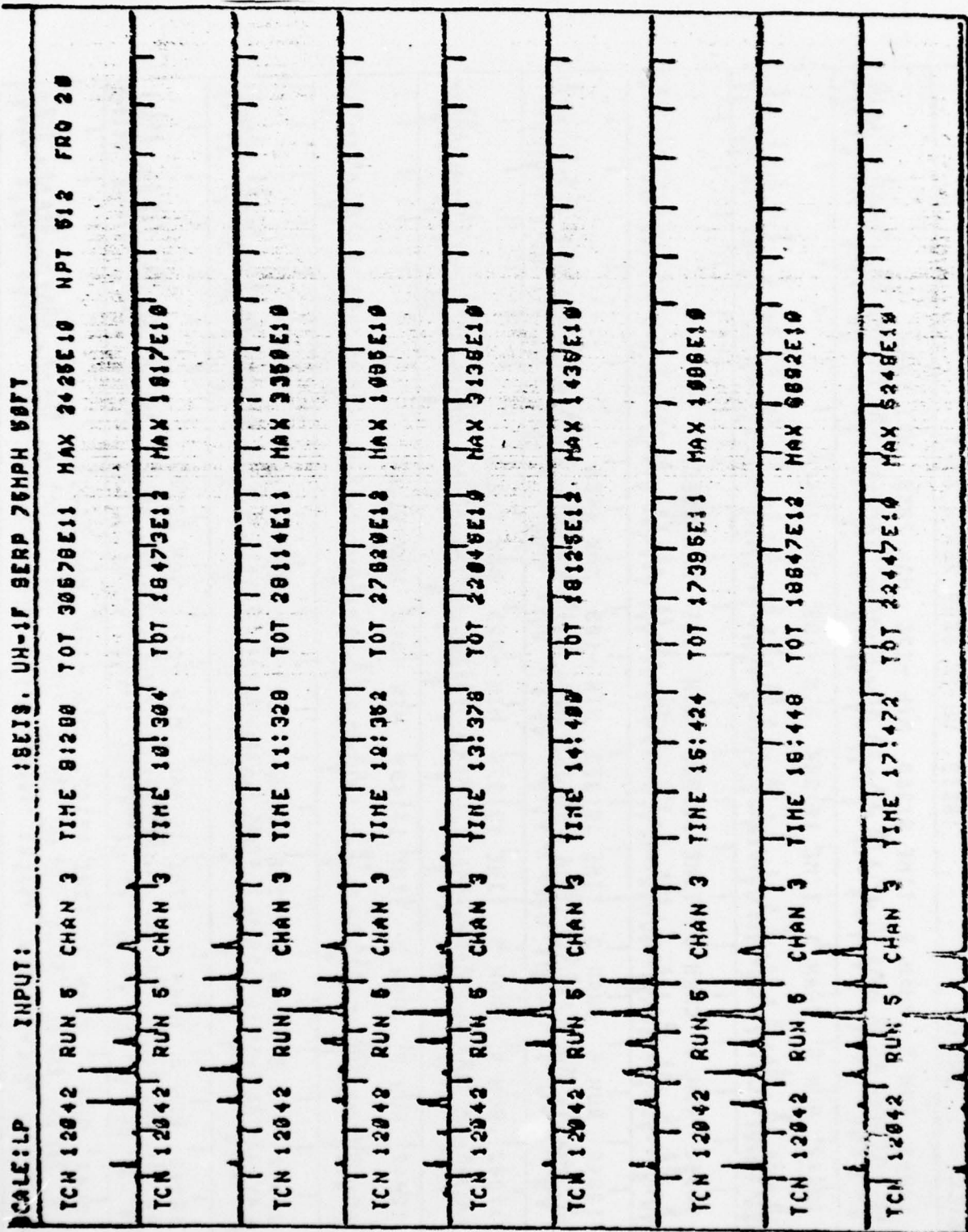


Figure 13-2 Helicopter Signature Frequency Domain

main rotor revolution speeds. With this intimate knowledge of the problem and its characteristics, the analyst can design features which will stand the test of time. It is highly unlikely that any statistically-based routine could ever have "learned" or "compensated" for the aforementioned characteristics.

The need to provide an efficient research tool for aiding a researcher in bridging the aforementioned gap between theory and practical applications in signal or waveform pattern recognition problems, has been the motivating force behind the design and implementation of WPS. It is hoped that WPS fulfills that need.

#### 13.2. RECOMMENDATIONS

It is felt that the Waveform Processing System is a useful and well-designed system. A wide usage of WPS will not only prove its worth but will also aid in its future development. Although a considerable amount of effort went into exercising and debugging the operational WPS software, it will only be through the extensive use of WPS that subtle errors, human factors deficiencies, and desirable additions to WPS, will be uncovered. Thus, while the capabilities of WPS are great, it will only be through the daily use of WPS by many users that the system will eventually reach its true potential.



#### REFERENCES

1. Sammon, J.W., "The On-Line Pattern Analysis and Recognition System - OLPARS", RADC-TR-68-263.AD#675 212.
2. Sammon, J.W. and Sanders, J.H. "The Waveform Analysis and Vector Extraction System - WAVES", Unsolicited Proposal to RADC May 1969.
3. Opitz, B. et al., "Feature Definition and Extraction" (IFES), RADC-TR-70-179.AD#876 134.
4. Sammon, J.W. et al., "The Discrete Variable Analysis and Classification System", Final Report to Contract F30602-70-C-0228, to be published.
5. Foley, D.H. et al., "Waveform Processing System (WPS)", to be published.
6. Sammon, J.W. et al., "Handprinted Character Recognition Techniques", RADC-TR-70-206.AD#876 875.
7. Klingbail, K.N., Connell, D.B., "Multics OLPARS Operating System," PAR Report No. 74-25, Final Report for Contract F30602-73-C-0352, April 1975.
8. Sammon, J.W., "Interactive Pattern Analysis and Recognition," IEEE Transactions on Computers, Vol. 19, July 1970.
9. Foley, D.H., "The Probability of Error on the Design Set as a Function of Sample Size and Dimensionality," RADC-TR-71-171.AD#738 648.
10. Toussaint, G.T., "Bibliography on Estimation of Misclassification," to be published in IEEE Transactions on Information Theory.